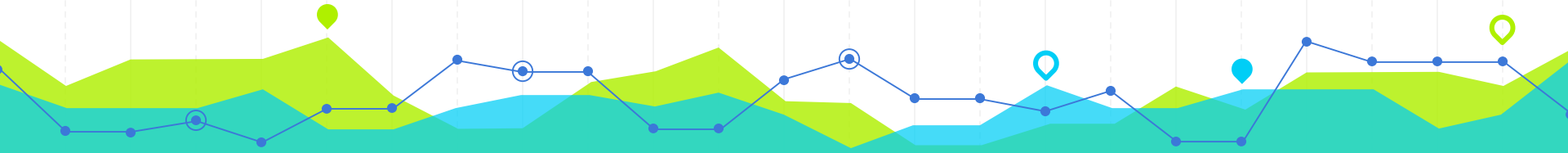
Higher-order Networks for Anomaly Detection

2nd KDD Workshop on Anomaly Detection in Finance



Presenters: Mandana Saebi and Jian Xu
Ph.D. Advisor: Nitesh Chawla

iCeNSA, University of Notre Dame

Research by

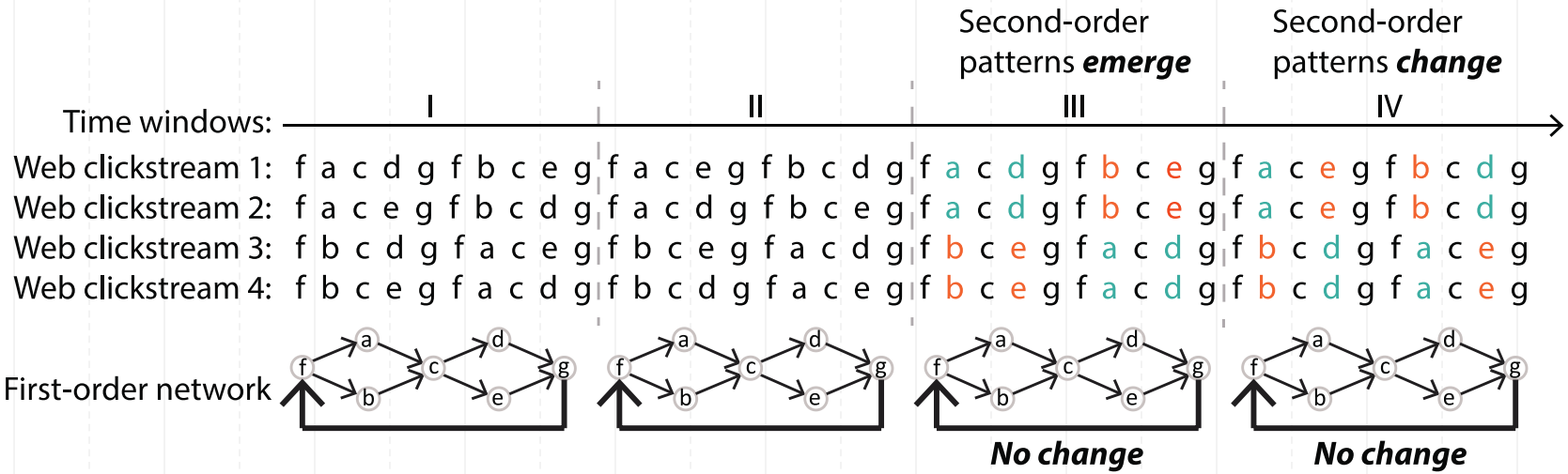
- **Mandana Saebi**, Ph.D. Candidate at University of Notre Dame
- **Jian Xu**, Ph.D. from University of Notre Dame, currently Data Scientist at Citadel
- **Nitesh Chawla**, University of Notre Dame
- **Bruno Ribeiro**, Purdue University
- **Lance Kaplan**, Army Research Lab

Anomaly detection and higher-order patterns

- Anomalies are deviations from the expected behavior of a complex system
- It is important that the data representation does not lose important information
- Current network-based anomaly detection methods use the First-Order Network (FON) to represent the underlying raw data
- What if we miss anomalies that are only discoverable through **higher-order patterns**?

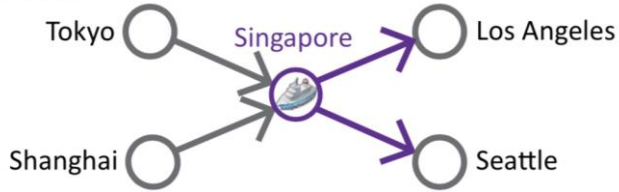
Example: Clickstream data

Anomalies completely hidden in FON



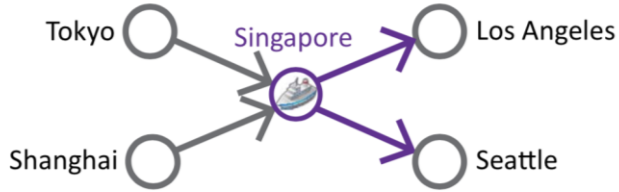
Higher-Order Network

First-order network



Higher-Order Network

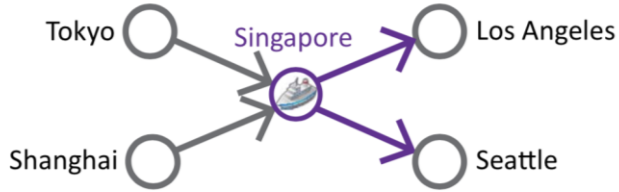
First-order network



Higher-order network (HON)

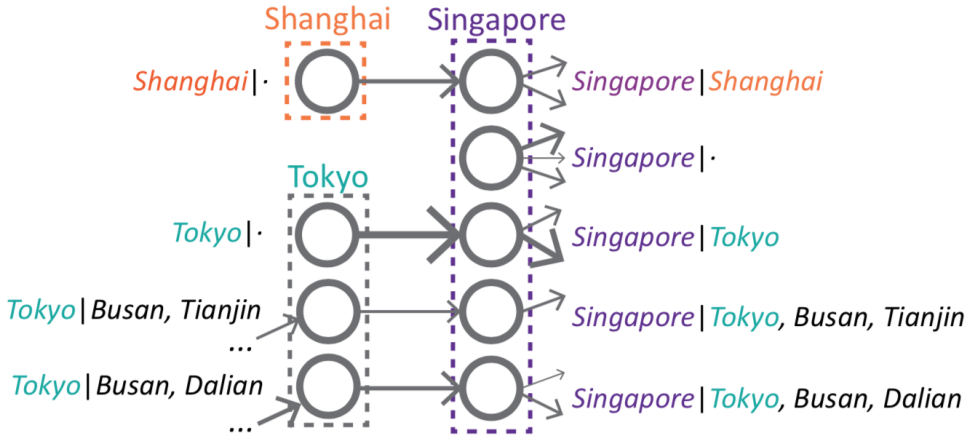
Higher-Order Network

First-order network



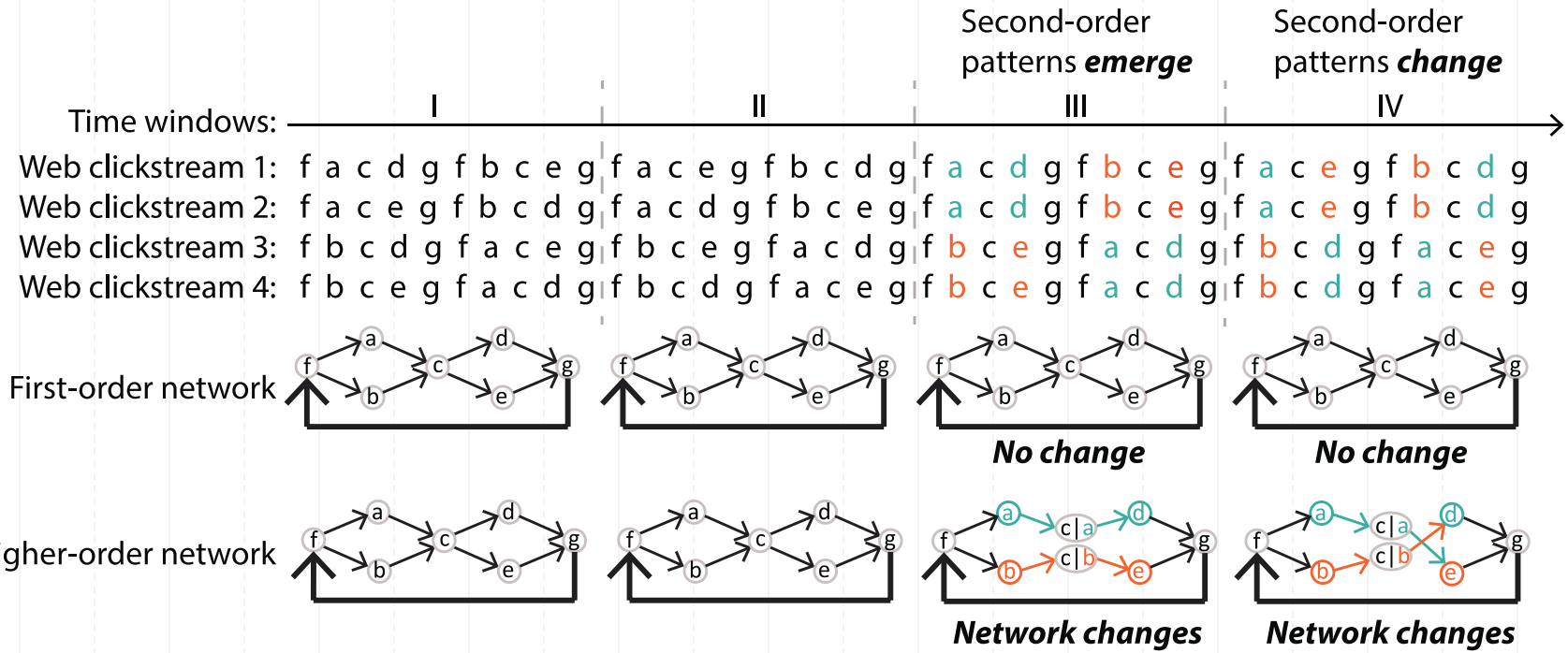
Higher-order network (HON)

Variable orders of dependencies in HON

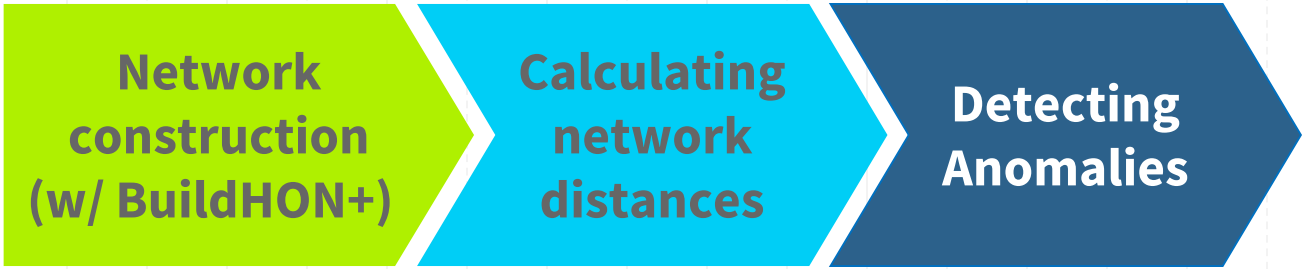


Example: Clickstream data

Anomalies revealed by representing data as HON



Anomaly detection in dynamic networks



BuildHON+

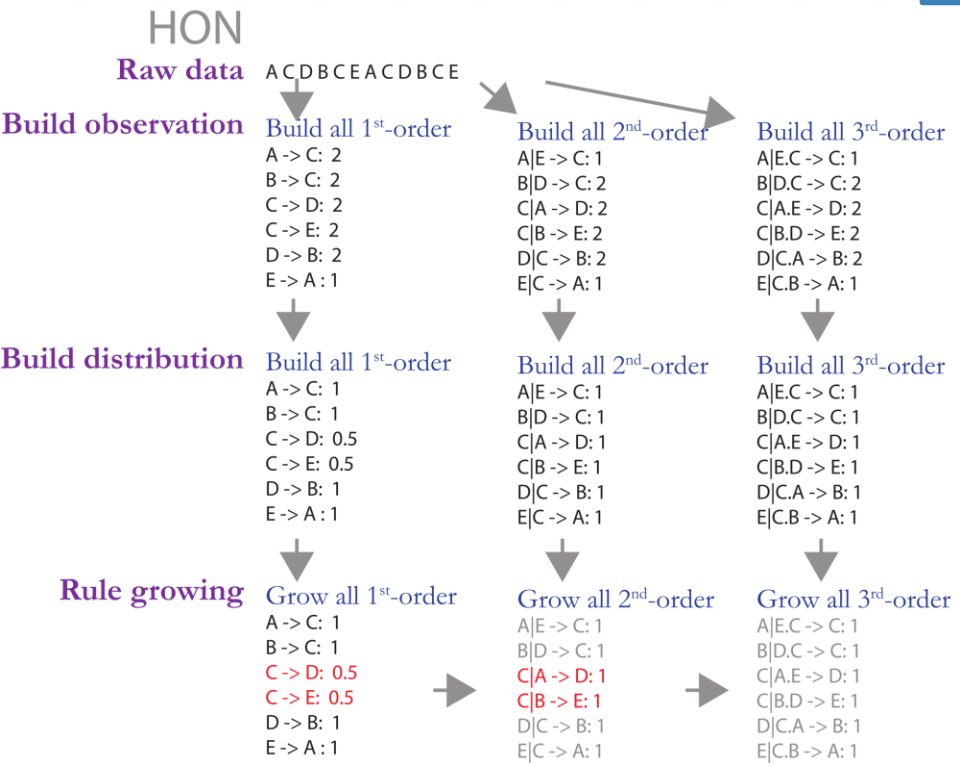
Compared to the original HON construction algorithm*, BuildHON+ is

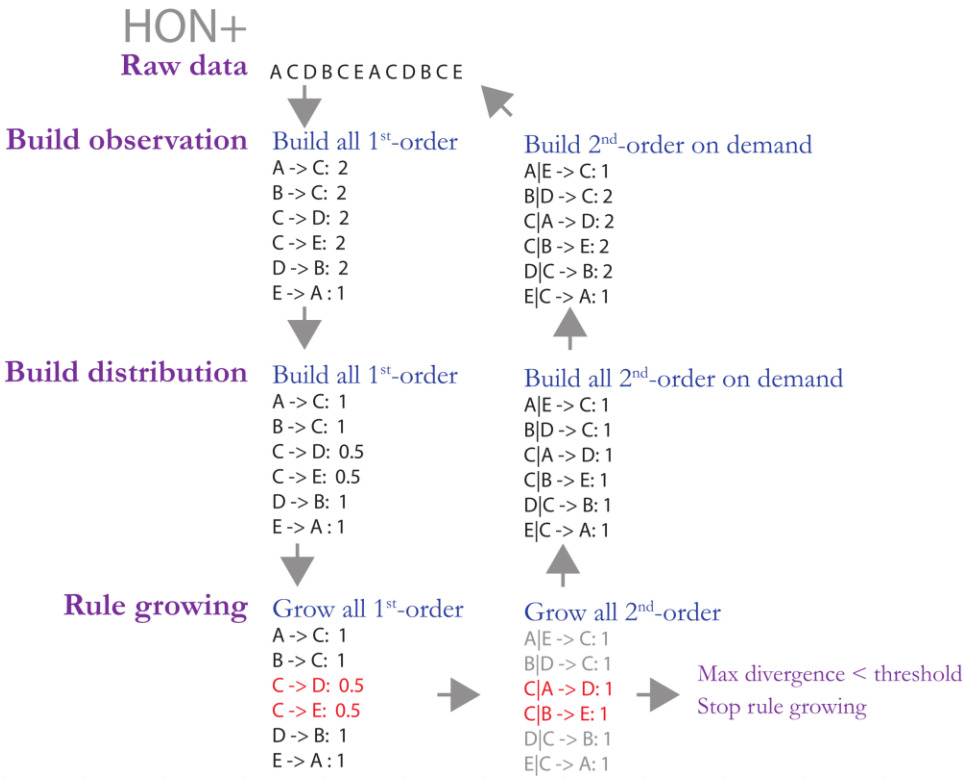
- Parameter-free
- More effective finding higher-order patterns (order 10 or higher)
- A magnitude faster with early stopping heuristics
- Python package available**

* Xu, Jian, Thanuka L. Wickramarathne, and Nitesh V. Chawla. "Representing higher-order dependencies in networks." *Science advances* 2, no. 5 (2016): e1600028.

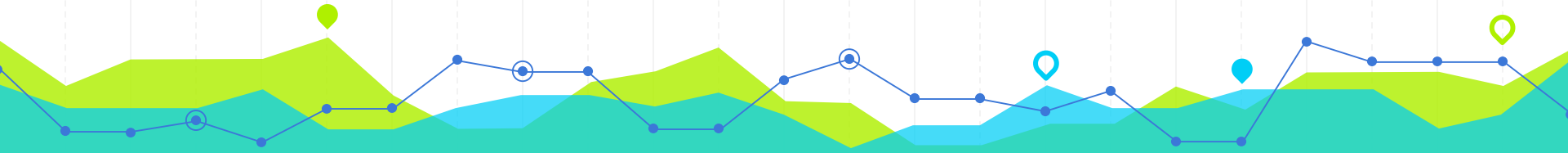
** https://github.com/xyjprc/hon

BuildHON+ has smaller search space for higher-order patterns



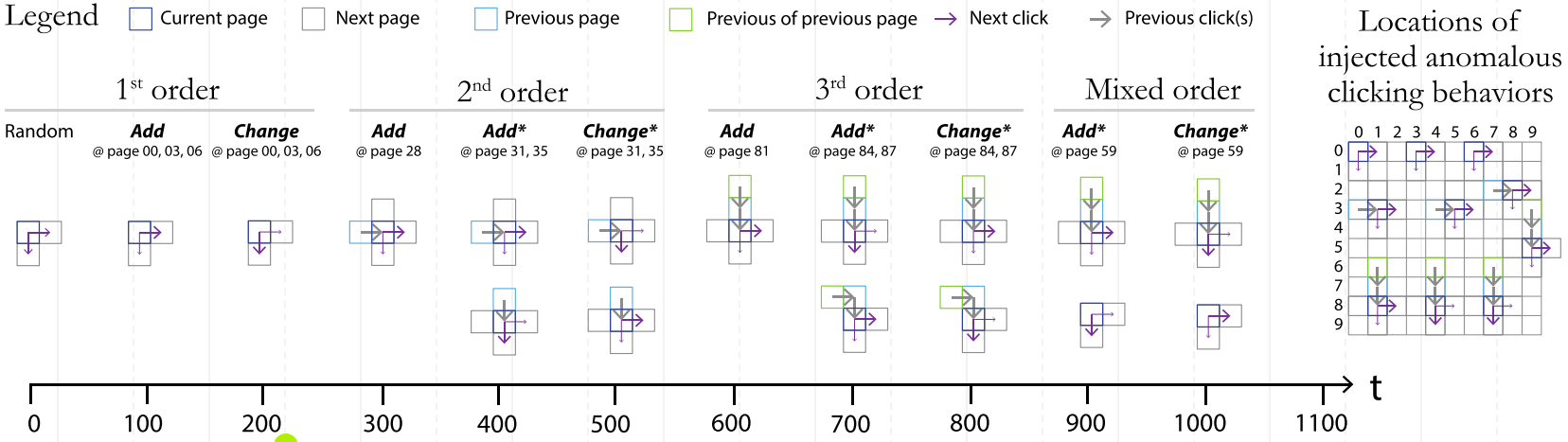


Results



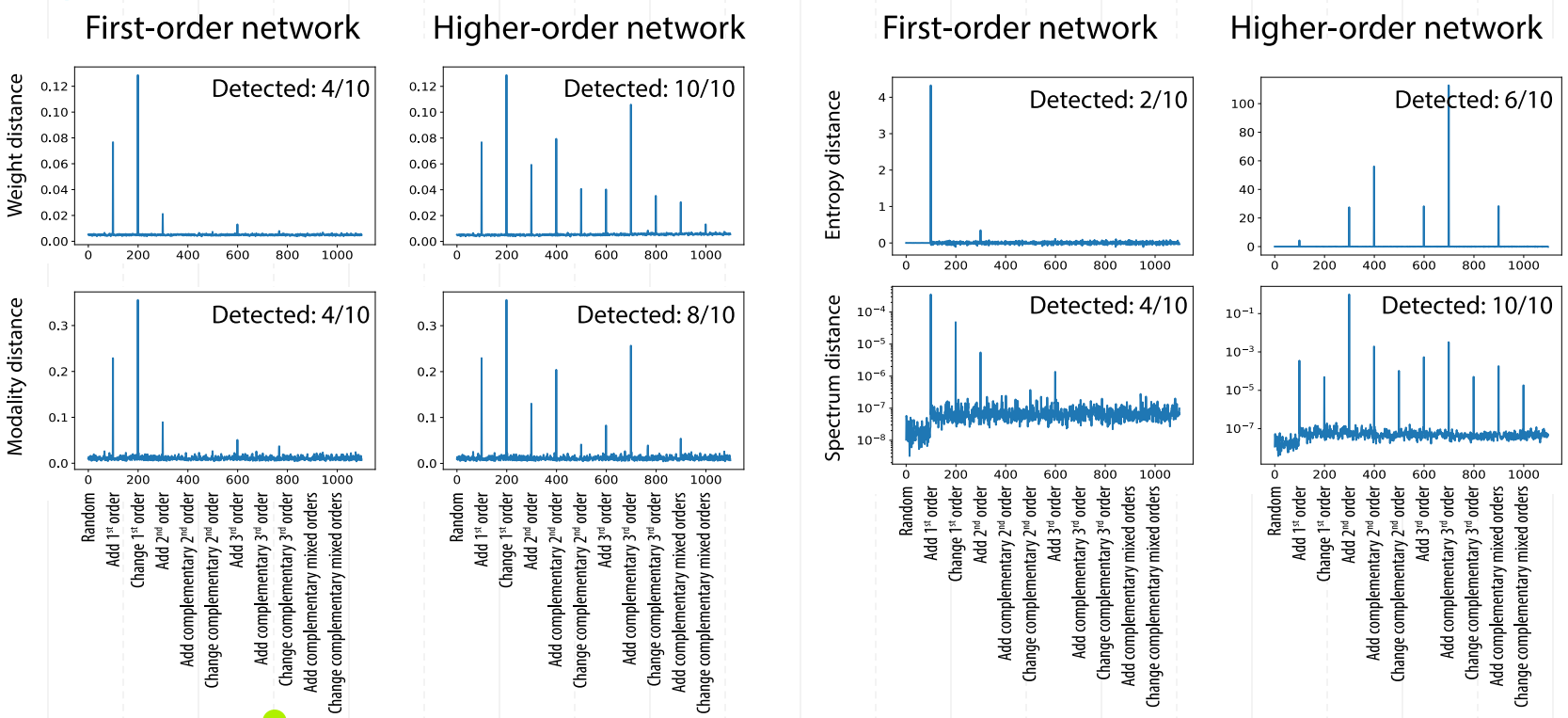
Synthetic data creation

- Web clickstream of 100,000 users navigating through 100 webpages, organized as a 10x10 grid
- Users change navigation behavior every 100 time windows
- Ground truth: 11 known anomalies of different natures (change of patterns of various orders)
- 11,000,000,000 clicks for anomaly detection task



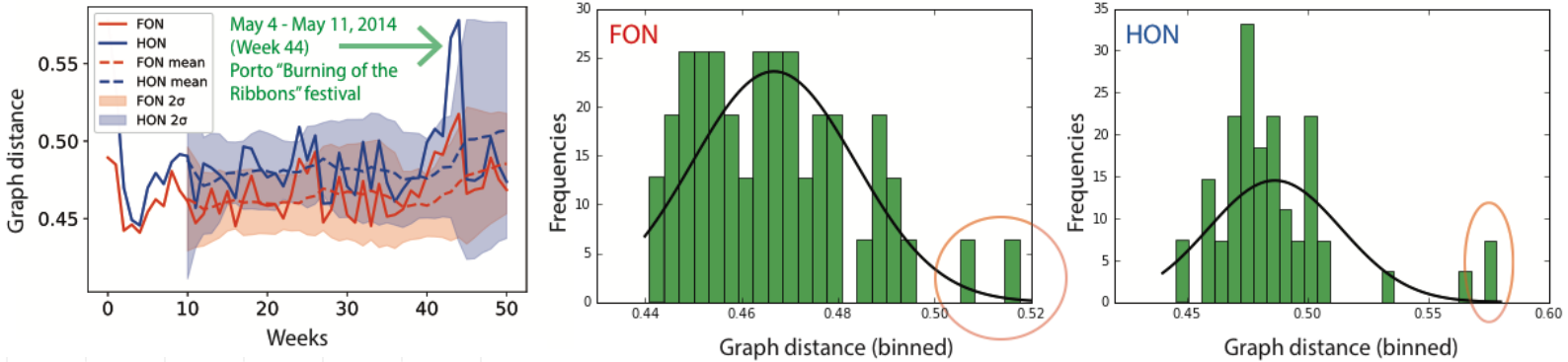
Synthetic data results

HON identifying more types of anomalies than FON



Real-world taxi data of Porto

- One year of all the 442 GPS trajectories from taxis in Porto, Portugal.
- Construct the FON and HON traffic networks for each week
- Compute the graph distances for neighboring time windows

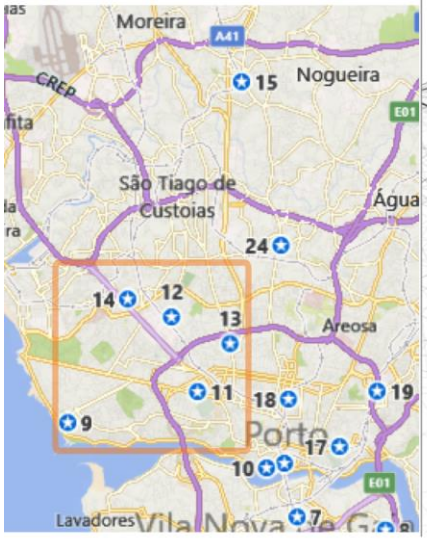


Anomalous traffic pattern more pronounced on HON

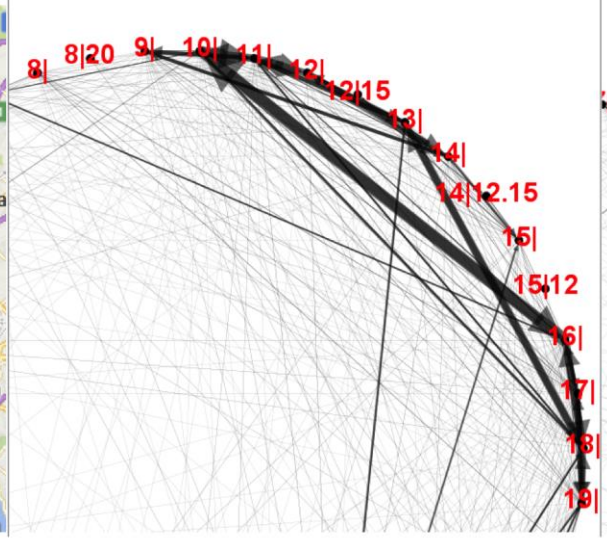
Network differences

HON highlighting differences in Location 9-14
Which are the main venues for the event

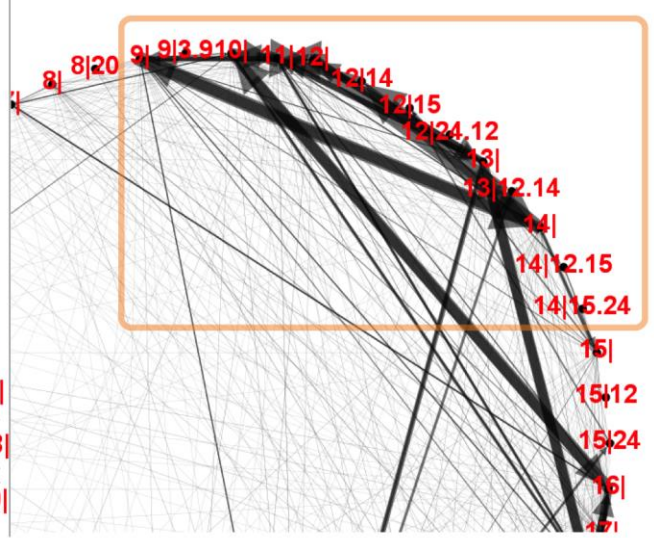
Police stations at Porto



Week 43



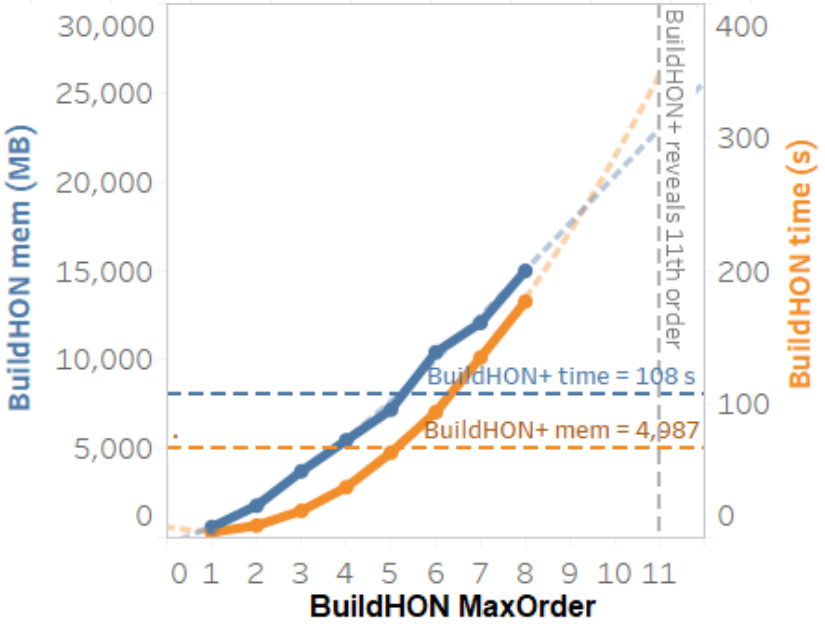
Week 44



Performance improvement:

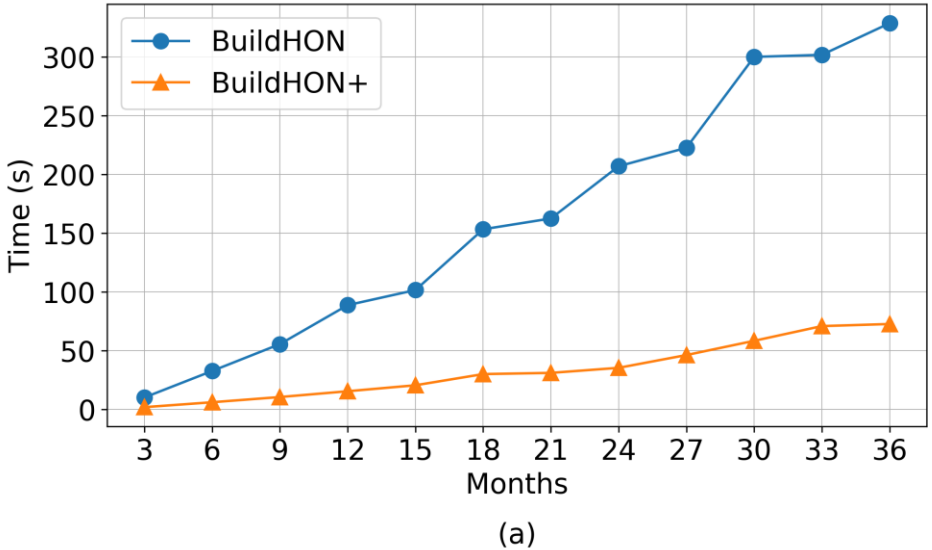
- **Data:** The shipping trajectories data with 3,415,577 voyages made by 65,591 ships, shown to have more than fifth order of dependencies.

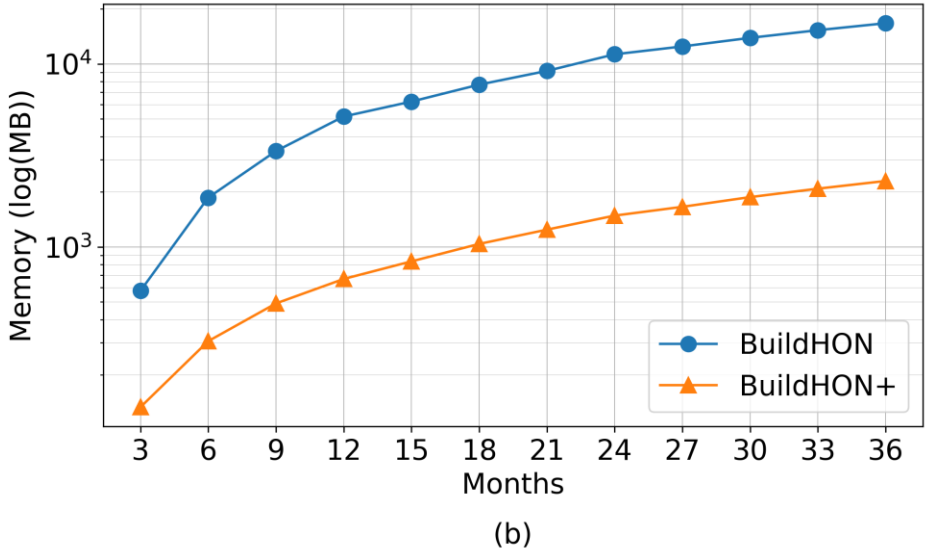
- **`BuildHON`** would need 3x time and 5x memory than **`BuildHON+`** to achieve the same results



Performance improvement:

BuildHON+ showing consistent advantage with various sized data





Summary

- There are anomalies that are only discoverable through **higher-order patterns**
- The higher-order network (HON) representation can help reveal such anomalies from sequential data
- BuildHON+ is a scalable HON construction algorithm
- HON-based dynamic network anomaly detection method applies to a wide variety of contexts

- More info: www.HigherOrderNetwork.com

Thanks!