A Framework for Mining Sequential Patterns from Spatiotemporal Event Data sets.

--Yan Huang, Liqin Zhang, Pusheng Zhang

Vijay Borra and Anuj Karpatne
Problem Statement

• What are Sequential Spatio-temporal Patterns?
• Examples: Crime, Disease spread, Climate
• Previous approaches in literature related to the problem:
  – Market Basket Analysis (Notion of transactions)
  – Trajectory (Same object considered)
• Applications in earth science, epidemiology, ecology, climatology

Approach to the Problem
• Developing significance measures of patterns for distinguishing meaningful pattern from spurious ones.
• Designing efficient algorithms using the significance measure
Contributions

- Density Ratio and Sequence Index for measuring the significance of a sequential pattern.

- STS-Miner and Slicing-STS-Miner algorithms for obtaining significant patterns using Sequence Index.
Proposed Model: Some definitions

- Event \( e_1 \) 'follows' event \( e \)
- Neighborhood of event \( e \)
- Density = Number of occurrences of an event in the spatio-temporal domain
- DensityRatio \( (E, E') \) = avg. density of \( E' \) in \( N(E) \) compared to density of \( E' \) in the entire space
- Sequence Index \( (ABC) \) = Min( DensityRatio\( (AB) \), DensityRatio\( (TailEventSet(AB), C) \) )
- Sequence Index
  - \( > 1 \) implies follows
  - \( < 1 \) implies Repels
  - \( = 1 \) implies independent
STS-Miner Algorithm

• Depth-first expand in the sequence tree:
  – A Node is a sequence
  – An edge is a 'follow' relationship.
• Steps in the algorithm:
  – Start with an empty tree
  – Compute Sequence Index between leaf nodes and all possible event types.
  – If resulting node is non-significant, mark as terminal node.
  – Iterate recursively for the new non-terminal child node.
Slicing STS-Miner Algorithm

- Partitioning large dataset using temporal slicing.
- Process a slice and proceed forward in time.
- Problems in overlapping regions:
  - Duplicity
  - Broken Sequences
- Concept of Crossing Tail Event Queue
- Depth-first Expand in Sequence Tree using Crossing Tail Event Queue
- No intermediate pruning before processing last slice
Validation and Results

- Performance evaluations for both in-memory models and disk based processing
- Synthetic simulations performed by varying parameters such as:
  Sequence Index, average number of sequences in each pattern, average pattern size, number of patterns etc.
- For In-Memory model STS-miner is faster than Slicing-STS-Miner. For Disk models Slicing-STS-miner is faster than STS-miner.

Reason:
Overhead of maintaining common tree.
Large Dataset Slice STS-Miner is better. Better handling of I/O

Results on Real Data Sets: Climate Data Analysis:
High Tempe --> High Evaporation : Low Evap --> High Temperature
Low Evaporation --> High Temperature --> High Evaporation
Assumptions

- Continuous neighborhood (problems with seasonality)
- Binary values used for calculating density ratio
- Linear patterns discovered.

A → B1 → C → D

B2

- Significance of Density Ratio = 1
Suggested Improvements

- Graph-based model of a pattern
- Real-valued Density Ratio variant
- Dynamic neighborhood generation depending on space, time, event type pair
- Monte Carlo Simulations for DR = 1
- More refined complexity analysis