Temporal Outlier Detection in Vehicle Traffic Data

Xiaolei Li, Zhenhui Li, Jiawei Han, Jae-Gil Lee
Outline

1. Motivation
2. Anomaly Definitions
3. Algorithm
4. Experiments
5. Conclusion
Motivation

- Many moving object trajectories being tracked in **road networks**
  - E-ZPass system in major cities
  - GPS trackers
- Finding unusual events or **anomalies**
  - Useful for city traffic control
  - Homeland security
Outline

1. Motivation
2. Anomaly Definitions
3. Algorithm
4. Experiments
5. Conclusion
What kind of Anomalies?

- “An observation (or a set of observations) which appears to be inconsistent with the remainder of that set of data.”
- Anomalies:
  - (1) other points in the dataset
  - (2) temporal history of itself
  - (3) temporal history of other points in the dataset.
What Kind of Anomalies?

- Unusual street segments
  - Speed of vehicles
  - Load of traffic
  - Association with other features

Example: A normal street?
Example - Anomaly

![Graph of load over time for Road Segment X and similar/dissimilar road segments.](image)

- **Road Segment X**: The load pattern for Road Segment X is shown as a solid line.
- **Similar Road Segments**: The load patterns for similar road segments are shown as dotted lines.
- **Dissimilar Road Segment**: The load pattern for a dissimilar road segment is shown as a dashed line.

The graphs illustrate the load variation over time from July 1 to July 5.
Example

- **University Ave.** usually has the similar traffic pattern as **Lincoln Ave.** and **Neil St.**
  - Similar speed patterns throughout the day

- On Nov. 30th, average speed on **University Ave.** drops from 40 to 25. University Ave. is an anomaly on that day?

- Might be two courses:
  - Traffic accident / road construction slows down traffic
  - Snow!
    - Global effect
    - Detect the similar change on Lincoln Ave. and Neil. St
Related work

- Compare with historical data of itself
- Compare with other data using time as a dimension on entire time span
Outline

1. Motivation
2. Anomaly Definitions
3. Algorithm
4. Experiments
5. Conclusion
Overall Framework

Traffic Data

Single Edges

Edge Sequences

Routes

Input Features And Similarity Fcn

Temporal Vector Construction

Temporal Similarity Vectors

1 2 3 ...

m

Temporal Anomaly Detection
Similar Neighbors

- On **day 1**, road **Y** is a similar neighbor of road **X**:  
  - X and Y have been similar **over several days** until day 1
- Question: how to maintain similar neighbors incrementally without looking back into history?
Temporal Neighbor Similarity Vector

- Maintain a vector for every edge
- Vector records similarity between the particular edge and all other edges in the road network
- Vector is updated at each time step
- Values in vector indicate **historical similarity**
- Drastic change in vector indicates **anomaly**

<table>
<thead>
<tr>
<th>Time Period 1</th>
<th>Time Period 2</th>
<th>Time Period 3</th>
<th>Time Period m</th>
</tr>
</thead>
</table>
| \[
S_{1,1}
\] | \[
S_{1,2}
\] | \[
S_{1,3}
\] | \[
S_{1,m}
\] |
| \[
S_{2,1}
\] | \[
S_{2,2}
\] | \[
S_{2,3}
\] | \[
S_{2,m}
\] |
| ... | ... | ... | ... |
| \[
S_{n,1}
\] | \[
S_{n,2}
\] | \[
S_{n,3}
\] | \[
S_{n,m}
\] |
Temporal Neighbor Similarity Vector

<table>
<thead>
<tr>
<th>Time Period 1</th>
<th>Time Period 2</th>
<th>Time Period 3</th>
<th>Time Period m</th>
</tr>
</thead>
</table>
| \[
\begin{bmatrix}
S_{1,1} \\
S_{2,1} \\
\vdots \\
S_{n,1}
\end{bmatrix}
\] | \[
\begin{bmatrix}
S_{1,2} \\
S_{2,2} \\
\vdots \\
S_{n,2}
\end{bmatrix}
\] | \[
\begin{bmatrix}
S_{1,3} \\
S_{2,3} \\
\vdots \\
S_{n,3}
\end{bmatrix}
\] | \[
\begin{bmatrix}
S_{1,m} \\
S_{2,m} \\
\vdots \\
S_{n,m}
\end{bmatrix}
\] |

1) How to update?  
2) How to detect anomaly?
Temporal Neighbor Similarity Vector

\[ \begin{bmatrix} s_{1,1} \\ s_{2,1} \\ \vdots \\ s_{n,1} \end{bmatrix} \quad \begin{bmatrix} s_{1,2} \\ s_{2,2} \\ \vdots \\ s_{n,2} \end{bmatrix} \quad \begin{bmatrix} s_{1,3} \\ s_{2,3} \\ \vdots \\ s_{n,3} \end{bmatrix} \quad \begin{bmatrix} s_{1,m} \\ s_{2,m} \\ \vdots \\ s_{n,m} \end{bmatrix} \]

**Intuition:**

<table>
<thead>
<tr>
<th>Historical Similarity</th>
<th>Day i's similarity</th>
<th>Updated Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>High</td>
<td>Increase lightly</td>
</tr>
<tr>
<td>High</td>
<td>Low</td>
<td>Decrease heavily</td>
</tr>
<tr>
<td>Low</td>
<td>High</td>
<td>Increase heavily</td>
</tr>
<tr>
<td>Low</td>
<td>Low</td>
<td>Decrease lightly</td>
</tr>
</tbody>
</table>

1) How to update?
Update Intuitions (Example)

Similarity

Time

Incremental similarity decreases

Sharp decrease For dissimilarity

similar

dissimilar
Update Rules

\[
\text{reward}(i, j, d) = \alpha_1 v_{i,j}^{d-1} - \alpha_2 \\
\alpha_1 < 1.0, \quad \alpha_2 \geq 0
\]

\[
v_{i,j}^d = v_{i,j}^{d-1} + \text{reward}(i, j, d)
\]

\[
\text{penalty}(i, j, d) = \beta v_{i,j}^{d-1} \\
\beta > 1.0
\]

\[
v_{i,j}^d = v_{i,j}^{d-1} - \text{penalty}(i, j, d)
\]
Anomaly Detection

- Measure anomaly by calculating similarity between Temporal Neighbor Similarity vector and daily neighbor vector

\[ OS(i, d) = \sum_{j=1, j \neq i}^{N} |v_{i,j}^d - v_{i,j}^{d-1}| \]

- Rank daily anomalies by dissimilarity amount
- Report top \( k \) anomalies every day
Outline

1. Motivation
2. Anomaly Definitions
3. Algorithm
4. Experiments
5. Conclusion
Experiments

- **Data:**
  - 24 days of moving taxicab data in San Francisco area in July, 2006
  - 800,000 separate trips; 33 million road segment traversals
  - 100,000 distinct road segments
Anomaly Example

![Graph of Load and Speed](image)

Load

- Outlier Road Segment
- Similar Neighbors Average

Time: July 3 to July 13

Speed

- Outlier Road Segment
- Similar Neighbors Average

Time: July 3 to July 15
Non-Anomaly Example
Efficiency

![Graph showing the relationship between runtime and number of days processed for different neighborhood radii.]

- **Neighborhood Radius = 5** (solid line)
- **Neighborhood Radius = 10** (dashed line)
Outline

1. Motivation
2. Anomaly Definitions
3. Algorithm
4. Experiments
5. Conclusion
Conclusion

- Contribution:
  - A new temporal outlier detection definition.
    - General outlier detection idea
  - Incremental outlier detection
    - Update similarity vector incrementally

- Further improvement:
  - How to evaluate outliers?
  - How to solve the sparse data problem?