Faculty of Electrical Engineering
University of Belgrade

Outlier Detection in Sensor Networks

(Sheng, Li, Mao)

Bogdan Azarić, 11/3035
bogdan.azaric@gmail.com

Outline

- Introduction
- Problem formulation
- Existing solutions
- Histogram query
- $O(d, k)$ solution
- Performance evaluation

Introduction

- Wireless sensor network
- Outlier phenomenon
- Outlier detection

Wireless sensor networks

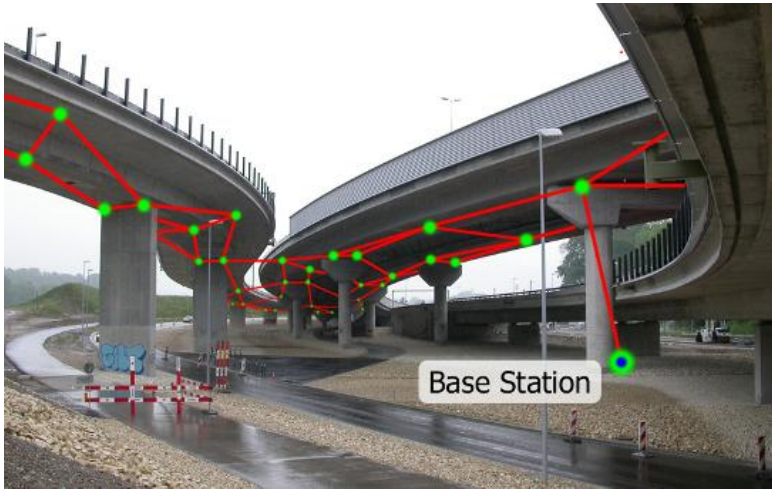
- Spatially distributed autonomous sensors
- Goal is to monitor physical or environmental conditions
  - Temperature
  - Sound
  - Presence of chemicals
- Consist of
  1. One or more sensors
  2. Communication radio
  3. Microcontroller
  4. Power supply

Outliers

- Values that are numerically different then the rest
- In our case values are sensor network readings
- Typically low dimension

Outlier detection

- Very high application potential
  - Construction
  - Medicine
  - Seismology
- Purpose is to detect anomalies
  - Wind induced bridge vibrations
  - Patient health condition change
  - Earthquake detection
- Types
  - Local outlier detection
  - Global outlier detection



Stress detecting wireless sensor network

Global vs. Local outlier detection

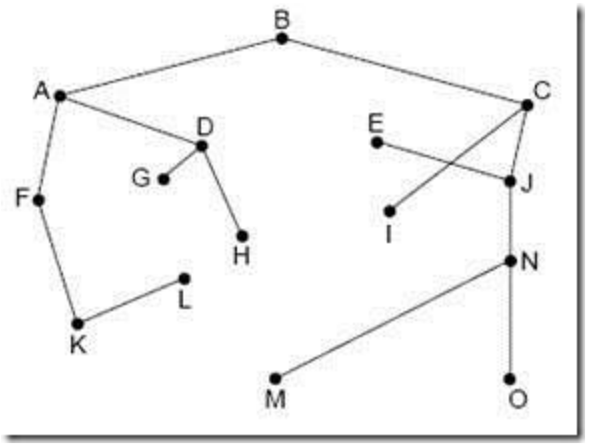
- Local
  - Only a small subset of data is examined
  - Detecting abnormal sensor readings in local proximity
  - Easy to locate by aggregating data
  - Example: surveillance monitoring
- Global
  - Whole network readings are examined
  - Very costly due to network-wide transmissions
  - Subject of this presentation

Problem formulation

- Outlier definition is based on K-th nearest neighbor
- $D^k(p) = |p_k - p|$
- Two most popular outlier definitions:
  1. *O(d, k) outlier if:* $D^k(p) > d$
  2. *O(n, k) outlier if there are no more than n-1 data points q such that* $D^k(q) > D^k(p)$
- Network consists of N nodes
- Routing tree rooted at the sink
- Data periodically generated
- Paremeters: *d* and *k* (def. 1), or *n* and *k* (def. 2)

Assumptions

- Routing tree topology robust
- Communication cost proportional to the packet size
- Each data point is represented by an integer



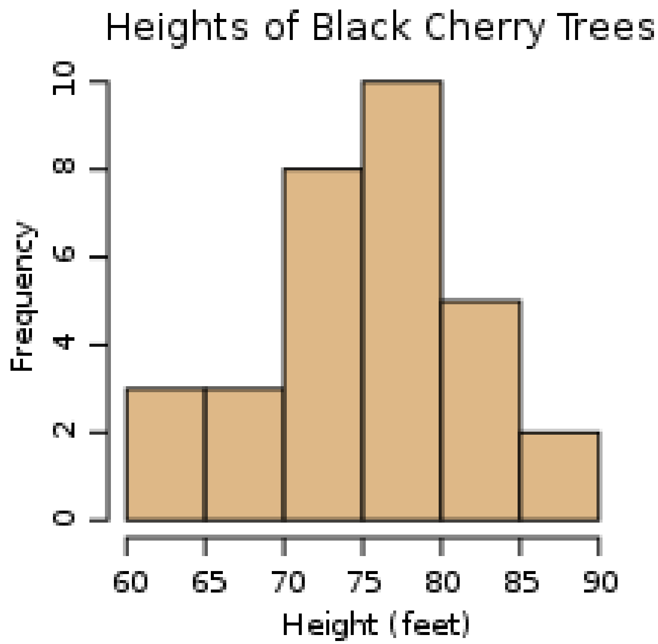
Routing tree example

Existing solutions

- Centralized scheme solution
  - All nodes sends all data poitns to the sink
  - Sink conducts the outlier detection
  - Drawbacks: huge communication cost
- J. Branch et. al. Solution
  - In-network scheme
  - Outcome is revealed to all sensors
  - Drawbacks: revealing outcome to all sensors costly
- S. Subramaniam et. al. solution
  - Keeps sliding window of the historic data
  - Drawbacks: huge memory consumption, may not reveal all

Histogram query

- Histogram: a rough estimate of the probability distribution
- We use equi-width histogram which is easy to aggregate
- Parameters:
  - Bucket width w
  - $max_i - min_i = w$
  - $min_i = max_{i-1}$
  - f_i = number of points in bucket i



Equi-width histogram example

Histogram query

- Goal is to calculate value pairs (l_i, u_i) for every bucket *i* such that for every point *p* in bucket *i*, $D^k(p) \in (l_i, u_i]$
- **Theorem 1**: if $f_i > k$, then $l_i = 0$ and $u_i = w - 1$, are lower and upper bounds for $D^k(p)$, where *p* is any data point in bucket *i*.
- **Theorem 2**:
  - we define a function: $F(t,i) = \sum_{j=i-t}^{i+t} f_j$
  - If $f_i \leq k$, we can find an integer $s \geq 0$ such that $F(s, i) \leq k$ and $F(s+1, i) > k$. Then $l_i = s \cdot w$, and $u_i = (s+2) \cdot w - 1$
- We utilize these theorems in our outlier detection schemes

Outlier detection for *O(d, k)*

- Outlier detection scheme for the first outlier definition:
  - *O(d, k) is outlier if:* $D^k(p) > d$
- Composed of multiple stages
  1. Obtain v_{min} and v_{max} histogram information
  2. Collect histogram
  3. Collect outliers and potential outliers
  4. Diffuse potential outliers and count the number of neighbors within *d*

Obtain v_{min} and v_{max}

- The first step of the O(d, k) outlier detection scheme
- The sink queries every node for its v_{min} and v_{max}
- At the end the sink has the total value range
- Every node sends at most $\log(v_{min} \cdot v_{max})$ bits of information

Collect histogram

1. The sink sets the global histogram parameters: v_{min}, v_{max}, w
   - Good value for bucket width w is d.
2. The sink sends a histogram query to all nodes
   - The query includes: w, v_{min}, v_{max}, and k
   - All non-leaf nodes send $\log(k \cdot d \cdot v_{min} \cdot v_{max})$ bits
3. Sensors divide histogram according to v_{min}, v_{max} and d
4. Sensors put all data points into one of the buckets
5. Hisogram is sent back to the sink
   - Histogram is aggregated every time it is sent upstream
   - Optimisation: if in each bucket we get more then $k + 1$ points we fix the counter for that bucket to $k + 1$
   - Communication cost per node: $(1 / d) \cdot \log(k + 1)$

Collect outliers and potential outliers

1. The sink applies Theorem 1 and Theorem 2
2. The sink analyses results for each bucket *i*
   - Case1: ui < d, all data points are non-outliers, they can be ignored
   - Case2: li ≥ d, all data points are outliers
   - Case3: otherwise, all data points are potential outliers
3. The sink sends a bit-vector query to collect all outliers and potential outliers
   - Query = $\{q_1, q_2, q_3 \ldots q_{[l/d]}\}$
   - Cost of points collection = $(N_o + N_{po}) \cdot \log(v_{max}) \cdot avgDist$

Diffuse potential outliers and count the number of Neighbors within d

1. Identifying some data points as outliers or non-outliers
2. For the rest, the sink sends queries comprised of list of potential outliers $\{p_1, p_2, p_3, \ldots\}$
3. Every sensor returns a list of summaries $\{f_1, f_2, f_3, \ldots\}$
   - The value of f_i is a number of points within distance d from p_i
   - Results are aggregated from children nodes to parent nodes
   - "$k + 1$" optimisation
4. The sink simply iterates over the result set and picks out every $f_i \leq k + 1$

Total communciation cost

- Total communication cost:
  - Each row represents one stage of the algorithm

$$C_{basic} = N \cdot \log(v_{\min} \cdot v_{\max}) +$$
$$+ N_{nl} \cdot \log(k \cdot d \cdot v_{\min} \cdot v_{\max}) + N \cdot \left\lceil \frac{l}{d} \right\rceil \cdot \log(k+1) +$$
$$+ N_{nl} \cdot \left\lceil \frac{l}{d} \right\rceil + (N_o + N_{po}) \cdot \log(v_{\max}) \cdot avgDist +$$
$$+ N_{nl} \cdot N_{po} \cdot \log(v_{\max}) + N \cdot N_{po} \cdot \log(k+1)$$

  - Where N_l is the number of non-leaf nodes
  - Drawbacks: if N_{po} is very large, collecting and difusing outliers will incur heavy cost
  - Soulution: enhanced scheme

Enhanced scheme

- More rounds of refined histogram queries can prune out additional data points
- Width of histogram bucket is now $w = d' < d$
- Each histogram query incurs additional communication cost
- Thus, the bucket width has to be carefully chosen

Performance evaluation

- Used real datasets from Intel Lab
- Data collected from 54 sensors during one month period
- 100 x 100 network, sensors randomly scattered
- Measurements for 1000 random topologies
- Two datasets of temperature measurements

Table 1: Network Setup

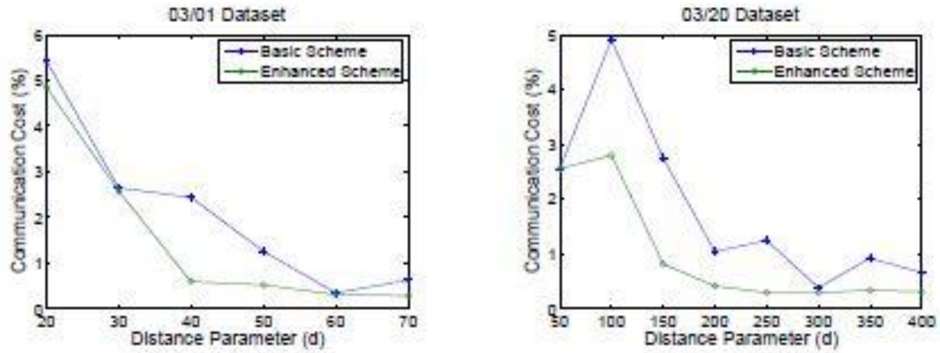
Number of Sensors(N)	54
Number of Non-leaf Nodes(N_{nl})	25.7
Radio Range	18
Avg. Hop Distance($avgDist$)	4.26

Table 2: Data Characteristics

	03/01 Dataset	03/20 Dataset
Number of Data Points	91468	76871
Maximum Value	3424	5008
Minimum Value	1499	363
Value Range	1926	4646

Performance evaluation

- 03/20 data has lot more outliers
- Evaluation in terms of total communication cost
- Comparing to the centralized scheme



- Conclusion: in worst case basic scheme consumes less than 5.5% of the cost of centralized scheme

References

- Outlier Detection in Sensor Networks (Bo Sheng, Qun Li, Weizhen Mao)
- Maimon O. and Rockach L. (Eds.) Data Mining and Knowledge Discovery Handbook
- Wikipedia - http://www.wikipedia.org/
- Online Outlier Detection in Sensor Data Using Non-Parametric Models

Thank you for your attention

Bogdan Azarić, 11/3035

bogdan.azaric@gmail.com