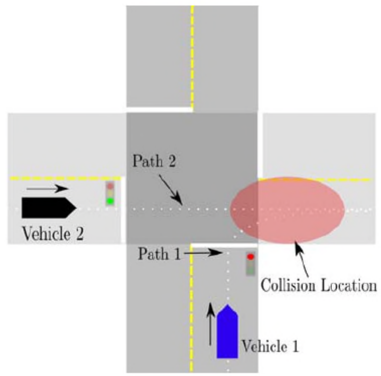
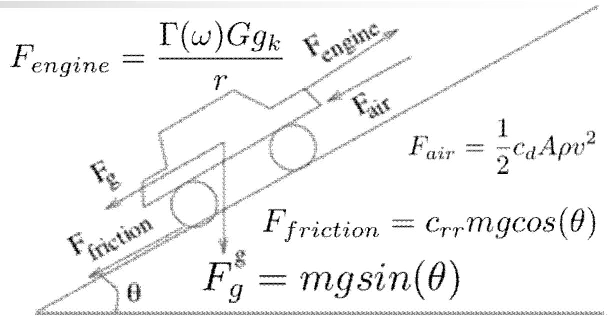
Sensing in Social Spaces

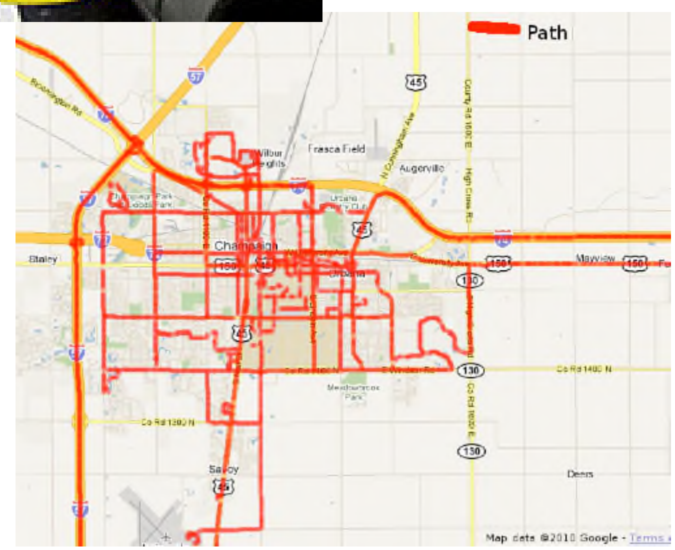
Project Ideas (Continued)

A Smart City Application: Sustainable Transportation





$$F_{car} = F_{engine} - F_{friction} - F_{air} - F_g$$



Transportation

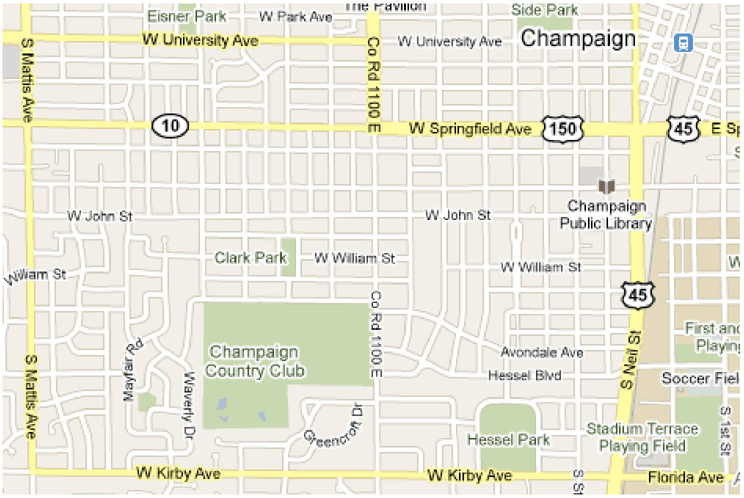
EPA Statistics (USA)

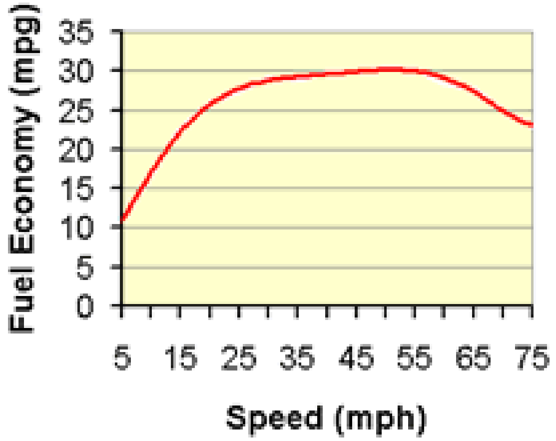
- 200 million light vehicles on the streets in the US
- Each is driven 12000 miles annually on average
- Average MPG is 20.3 miles/gallon
- **118 Billion Gallons of Fuel per year!**

- **Savings of 1% = One Billion Gallons**

GreenGPS: Fuel Efficient Vehicular Navigation

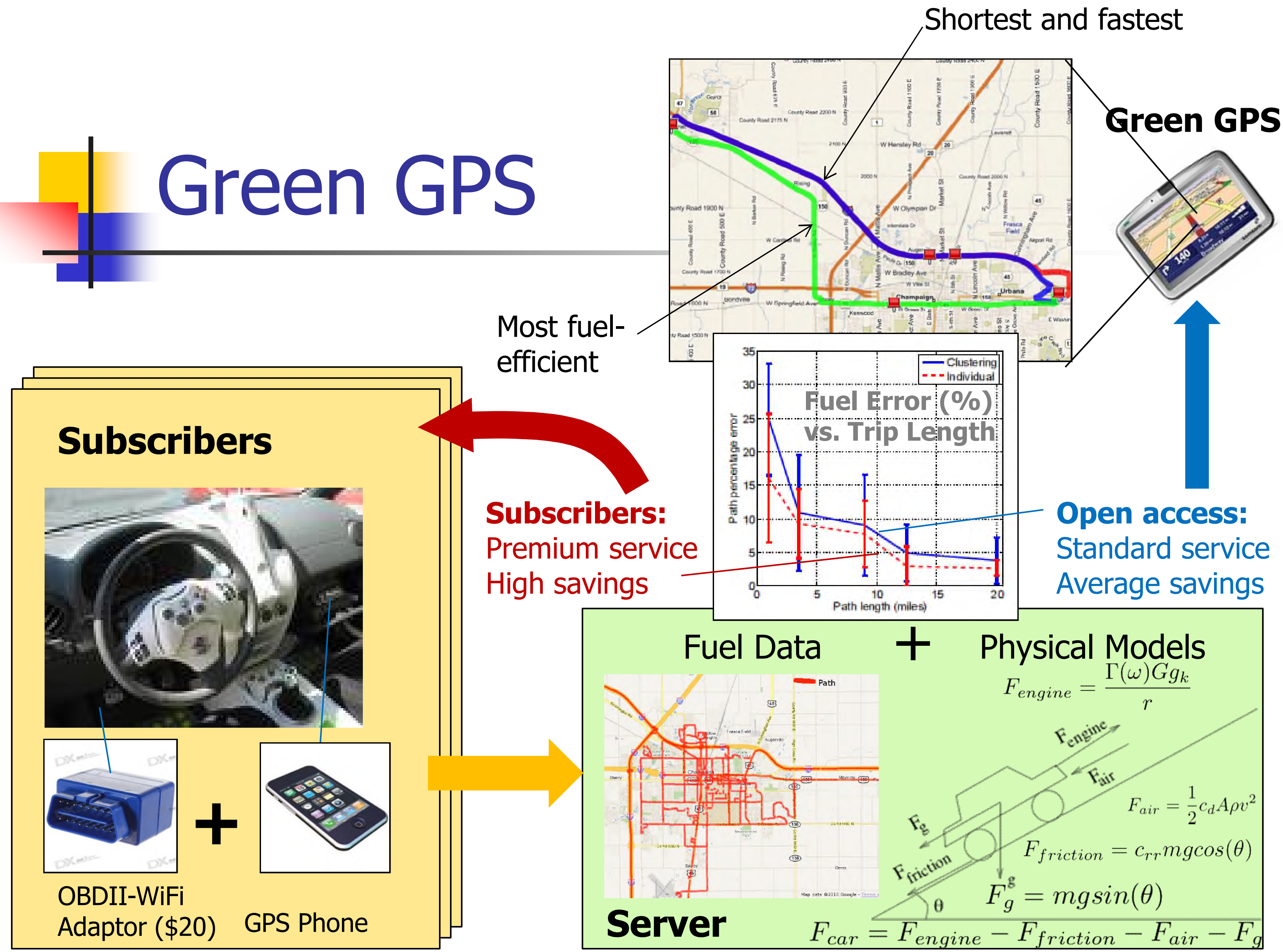
- Find the most fuel-efficient route (instead of a fastest or shortest)
- Fuel-efficient route is *different* from shortest or fastest route
  - Congestion → shortest may not be fuel efficient
  - MPG vs. speed is non-linear → fastest may not be fuel efficient





Source: US EPA

Green GPS

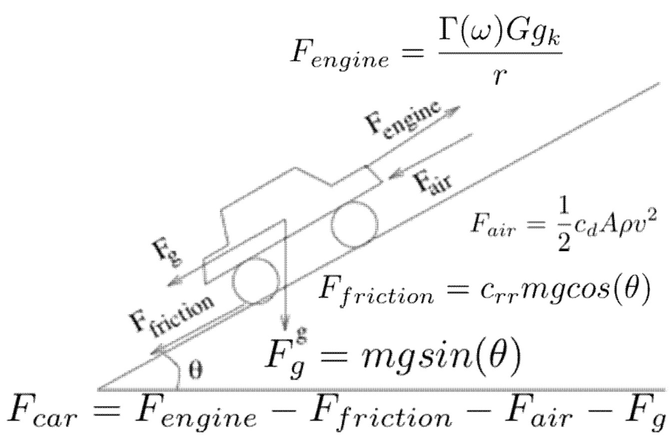


Shortest and fastest
Green GPS
Most fuel-efficient
Subscribers
OBDII-WiFi Adaptor ($20)
+
GPS Phone
Subscribers:
Premium service
High savings
Clustering
Individual
Fuel Error (%) vs. Trip Length
Path percentage error
Path length (miles)
Open access:
Standard service
Average savings
Fuel Data
+
Physical Models
Path
Server
$F_{engine} = \frac{\Gamma(\omega)Gg_k}{r}$
$F_{air} = \frac{1}{2}c_d A \rho v^2$
$F_{friction} = c_{rr} mgcos(\theta)$
$F_g = mgsin(\theta)$
$F_{car} = F_{engine} - F_{friction} - F_{air} - F_g$

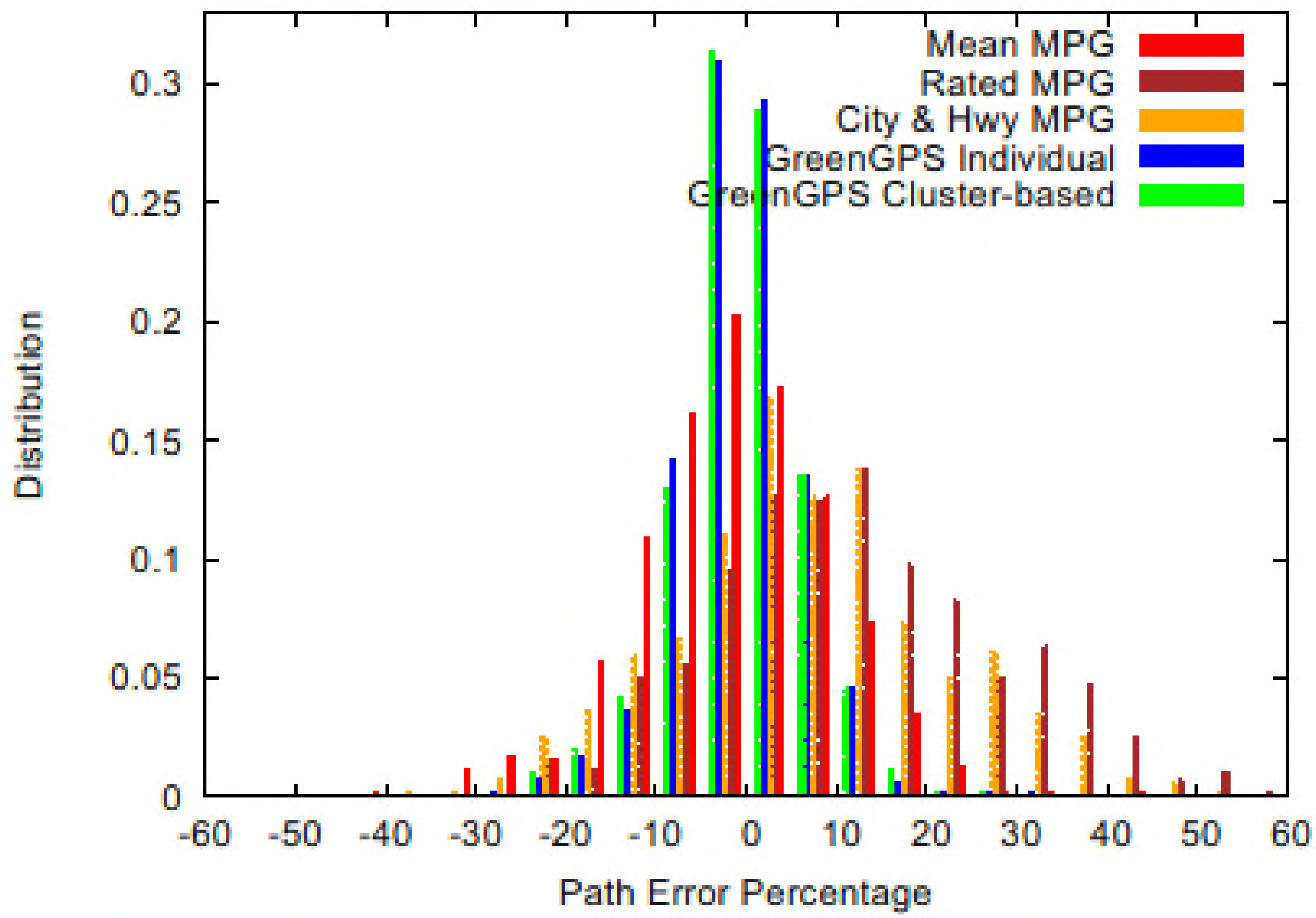
Fuel Consumption Model

- Simple model for fuel consumption derived from first principles
- The model is then is approximately recast in terms of easily measurable crowdsensed parameters (e.g., locations of stop signs, traffic lights, speed limits, and actual traffic conditions)

$$gpm = k_1 m \bar{v}^2 \frac{ST + \nu TL}{\Delta d} + k_2 m \frac{\bar{v}^2}{\Delta d} + k_3 m cos(\theta) + k_4 A \bar{v}^2 + k_5 m sin(\theta)$$



Error Distribution in Fuel Prediction



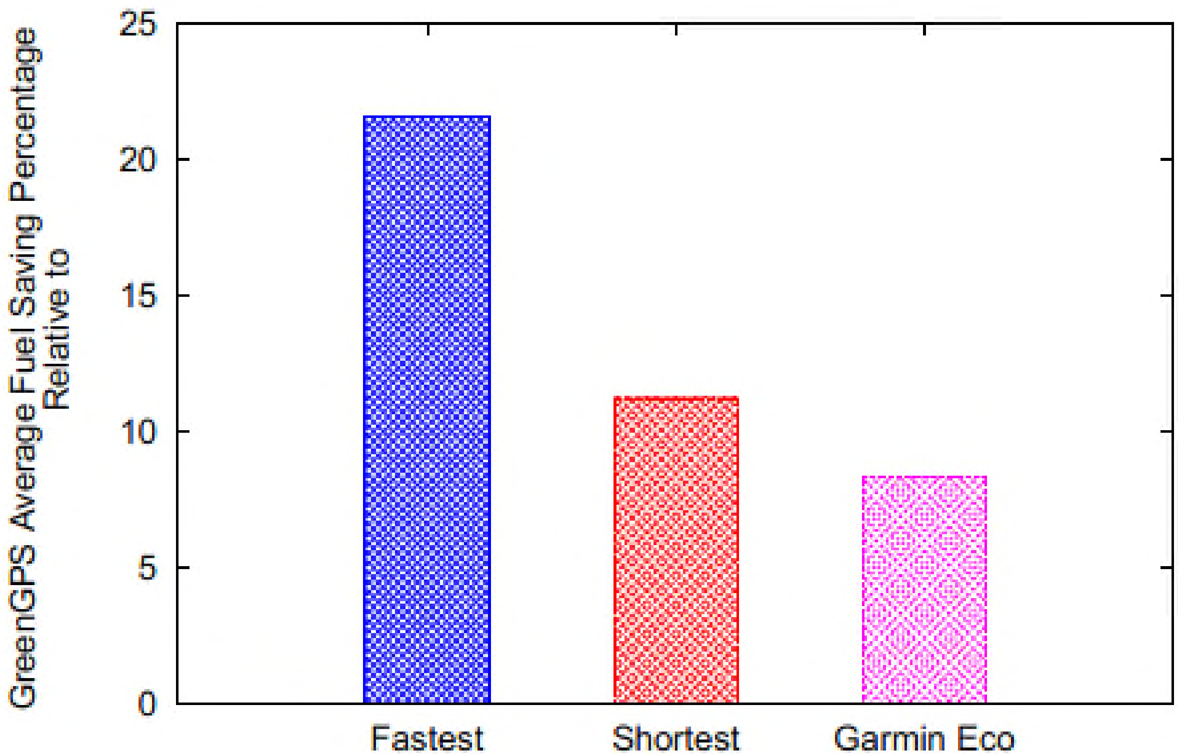
Fuel Consumption Examples

- Experiments on five cars, each does *four round-trips* between 2 landmarks in Urbana-Champaign on *fastest* and *shortest* routes, showing that neither wins consistently in being energy-optimal

Car	Route	Better Route	Difference
Honda Accord 2001	Home1 to Mall	Shortest	31.4%
	Home1 to Gym	Shortest	19.7%
Ford Taurus 2001	Home2 to Restaurant	Shortest	26%
Toyota Celica 2001	Home2 to Work	Fastest	10.1%
Nissan Sentra 2009	Home3 to Clinic	Fastest	8.4%
Honda Civic 2002	Home4 to Work	Fastest	18.7%

End Result: Fuel Savings

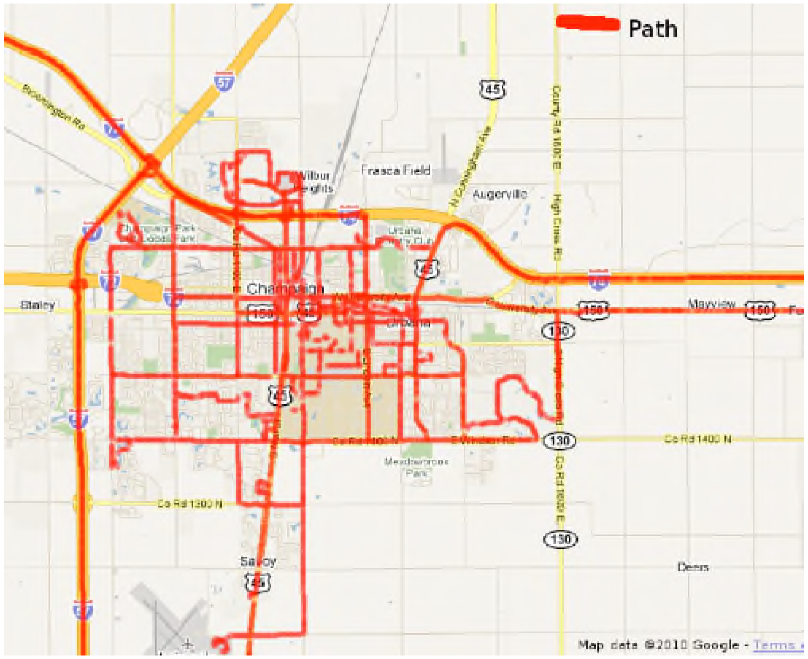
- The bottomline: percentage of fuel is saved over fastest, shortest, and GarminEco routes:



Crowdsensing challenge

Extrapolation from Sparse Data
(Conditions of Sparse Deployment)

Extrapolation from Sparse Data



Fuel consumption of
A few cars driven on a
few roads

Predict fuel consumption of
any car on any road

Generalization and Modeling

- Regression modeling:
  - Problem: one size does not fit all. Who says that Fords and Toyotas have the same regression model?
- Regression model per car?
  - Problem: Cannot use data collected by some cars to predict fuel consumption of others.
- Challenge: Must jointly determine both (i) regression models and (ii) their scope of applicability, to cover the whole data space within an acceptable modeling error.

Idea: Data Clustering
(Using Data Cubes)

- Data cubes are clustering technique that groups all crowd-sensed data according to several *alternative* dimensions (clustering policies) such as by car make, model, or year.
- A regression model is then derived for resulting clusters
- Different clustering policies are evaluated in terms of their fuel prediction error to determine the best policy
- When a navigation request from a new vehicle arrives:
  - The best clustering policy is used to add the vehicle to existing clusters
  - The regression model for this cluster is used to predict the vehicle’s fuel consumption

The Regression Cube Model

- Data cells correspond to:
  - Output attributes $Y_c = \{y_i\}$
  - Each associated with k input attributes $x_{i1}, \ldots, x_{ik}$, $X_c = \{x_{ij}\}$

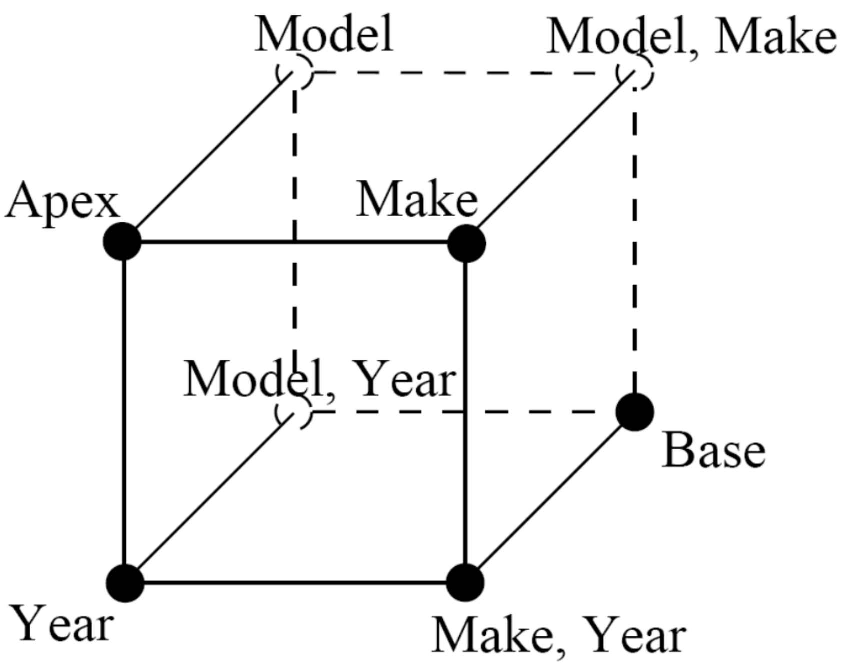
- Data cells store the following measures:
  - Regression model coefficients:

$$\hat{Y}_c = X_c \hat{\eta}_c$$

  - Regression modeling error:

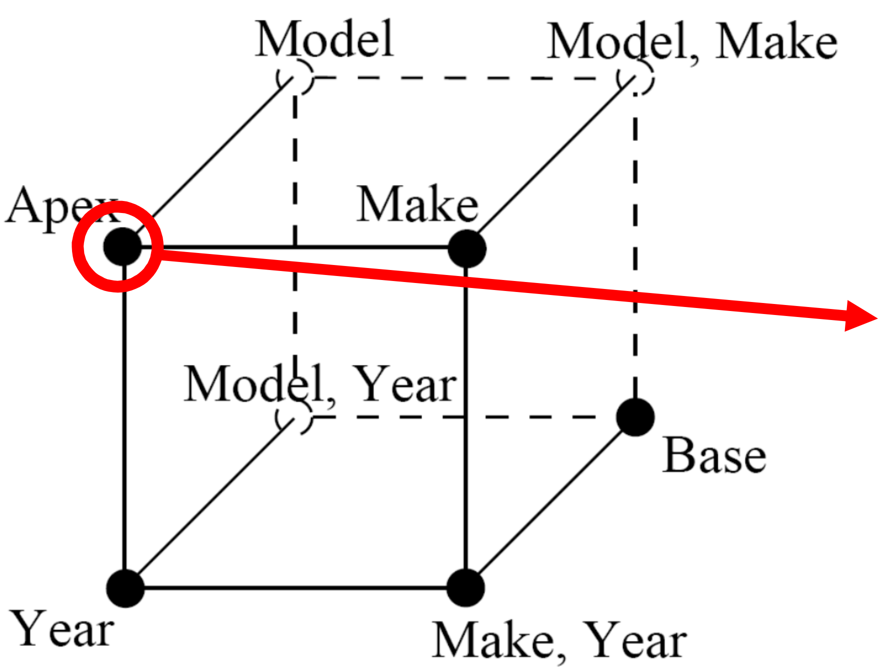
$$Err_c = (Y_c - X_c \hat{\eta}_c)^T (Y_c - X_c \hat{\eta}_c)$$

Example of Regression Cubes



- Goal: predict fuel consumption
  - Group by make, model, or year

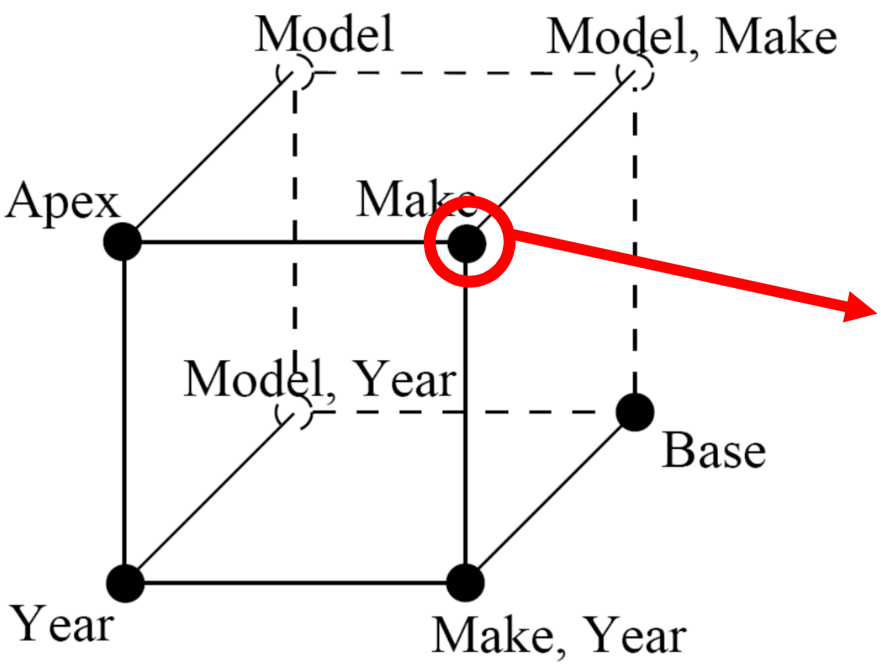
Example of Regression Cubes



Data Cells:

(*,*,*) – X, Y

Example of Regression Cubes



Data Cells:

(Toyota,*,*) – X_{c1} Y_{c1}
(Ford,*,*) – X_{c2} Y_{c2}
(Honda,*,*) – X_{c2} Y_{c3}

Data Cell Measures

- Main challenge: compute data cell measures recursively and without reprocessing raw data
- Measures can be classified as:
  - Distributive – $f(x_1, x_2, x_3) = f(f(x_1, x_2), x_3)$ - Efficient
    - Examples: sum, count
  - Algebraic/Compressible – An algebraic combination of distributive functions - Efficient
    - Example: average = sum/count
  - Holistic – Reprocess raw data - Inefficient
    - Example: median

The Challenge in Regression Cubes

- Main problem: Model parameters and estimation error are not distributive

$$\hat{Y}_c = X_c \hat{\eta}_c$$

$$Err_c = (Y_c - X_c \hat{\eta}_c)^T (Y_c - X_c \hat{\eta}_c)$$

An Efficient Representation

- Compressed representation of a cell c:
  - $\rho_c = Y_c^T Y_c$: scalar value
  - $\Theta_c = X_c^T X_c$: vector of size k
  - $\nu_c = X_c^T Y_c$: k by k matrix
  - n_c : number of samples

$$\rho_c = \sum_{i=1}^{m} \rho_i \qquad \nu_c = \sum_{i=1}^{m} \nu_i \qquad \Theta_c = \sum_{i=1}^{m} \Theta_i \qquad n_c = \sum_{i=1}^{m} n_{c_i}$$

- These matrices are distributive measures

An Efficient Data Cube for Fuel Consumption Regression Models

- Model coefficients:

$$\hat{\eta}_c = (X_c^T X_c)^{-1} X_c^T Y_c = \Theta_c^{-1} \nu_c$$

- Error:

$$\begin{aligned} Err_c &= (Y_c - X_c\hat{\eta}_c)^T (Y_c - X_c\hat{\eta}_c) = \\ &Y_c^T Y_c - (X_c\hat{\eta}_c)^T Y_c - Y_c^T X_c\hat{\eta}_c + (X_c\hat{\eta}_c)^T X_c\hat{\eta}_c = \\ &\rho_c - \hat{\eta}_c^T \nu_c - \nu_c^T \hat{\eta}_c + \hat{\eta}_c^T \Theta_c \hat{\eta}_c \end{aligned}$$

- Model coefficients and regression error are compressible measures

Idea #2: Model Reduction

- Independently find *the set of model parameters, L,* for each cell, such that:
  - The cell is reliable
  - Corresponding error is minimized
  - Challenge: Exponential number of *L*s

Attributes
Velocity (v)
Mass (m)
Frontal area (A)
Stop signs (S)

	Error	Reliable
L = {v}	0.031	yes
L = {m}	0.152	yes
L = {A}	0.043	yes
L = {S}	0.056	yes

Computing data Cell Confidence

- Measure of confidence:
  - Probability at which the actual coefficients are far from the estimate

$$Pr[||\hat{\eta_c} - \eta_c|| > \delta]$$

$$Pr[||\hat{\eta_c} - \eta_c|| > \delta] \leq \frac{k\sigma^2}{\delta^2 \lambda_{min}(X_c^T X_c)}$$

$$\hat{\sigma}^2 = \frac{Err_c}{n_c}$$

- Reliable Cell:

$$\frac{k\hat{\sigma}^2}{\delta^2 \lambda_{min}(\Theta_c)} < 0.05$$

Idea #2: Model Reduction

- Independently find *the set of model parameters, L*, for each cell, such that:
  - The cell is reliable
  - Corresponding error is minimized
  - Challenge: Exponential number of *L*s

	Error	Reliable
L = {v}	0.031	yes
L = {m}	0.152	yes
L = {A}	0.043	yes
L = {S}	0.056	yes

Attributes
Velocity (v)
Mass (m)
Frontal area (A)
Stop signs (S)

Idea #2: Model Reduction

- Independently find *the set of model parameters, L,* for each cell, such that:
  - The cell is reliable
  - Corresponding error is minimized
  - Challenge: Exponential number of *L*s

Attributes
Velocity (*v*)
Mass (*m*)
Frontal area (*A*)
Stop signs (*S*)

L = {v}
L = {m}
L = {A}
L = {S}

→

	Error	Reliable
L = {v, m}	0.021	no
L = {v, A}	0.030	yes
L = {v, S}	0.028	yes

Idea #2: Model Reduction

- Independently find *the set of model parameters, L,* for each cell, such that:
  - The cell is reliable
  - Corresponding error is minimized
  - Challenge: Exponential number of *L*s

Attributes
Velocity (*v*)
Mass (*m*)
Frontal area (*A*)
Stop signs (*S*)

L = {v}
L = {m}
L = {A}
L = {S}

→

	Error	Reliable
L = {v, m}	0.021	no
L = {v, A}	0.030	yes
L = {v, S}	0.028	yes

Idea #2: Model Reduction

- Independently find *the set of model parameters, L,* for each cell, such that:
  - The cell is reliable
  - Corresponding error is minimized
  - Challenge: Exponential number of *L*s

L = {v}
L = {m}
L = {A}
L = {S}

→

L = {v, m}
L = {v, A}
L = {v, S}

→

	Error	Reliable
L = {v, S, m}	0.024	no
L = {v, S, A}	0.026	no

Idea #2: Model Reduction

- Independently find *the set of model parameters, L,* for each cell, such that:
  - The cell is reliable
  - Corresponding error is minimized
  - Challenge: Exponential number of *L*s

L = {v}
L = {m}
L = {A}
L = {S}

→

L = {v, m}
L = {v, A}
L = {v, S}

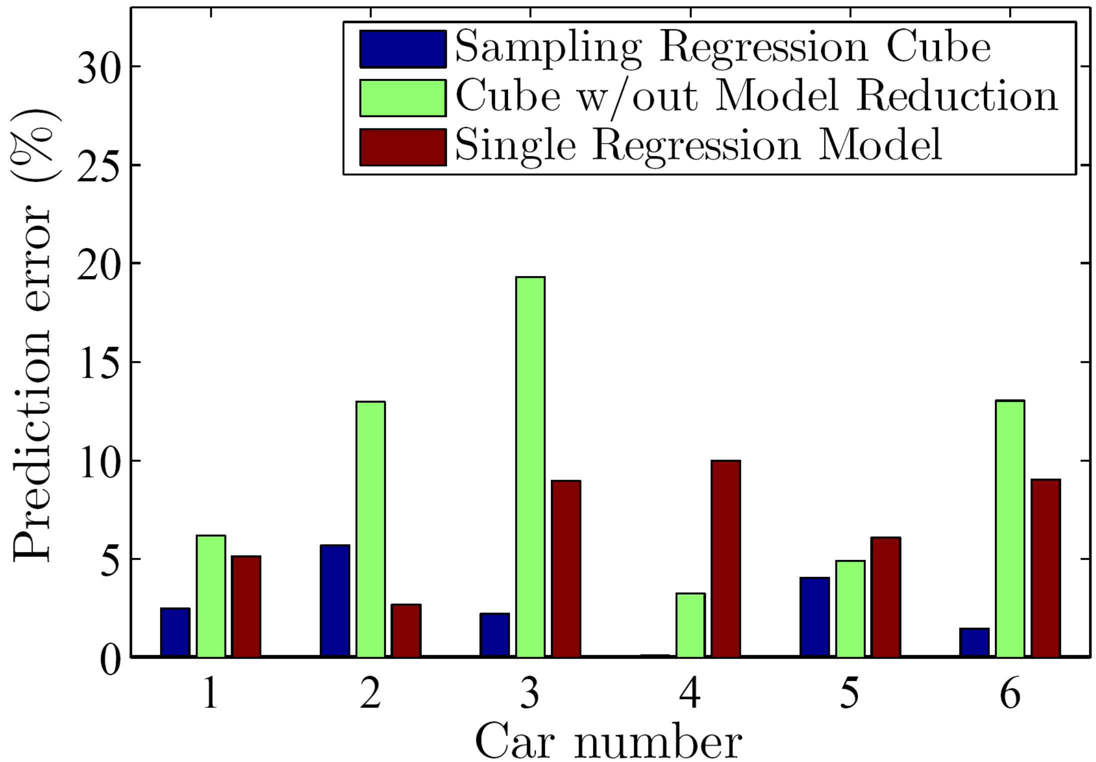
→

L = {v, S, m}
L = {v, S, A}

Reduced Model: {v, S}

Accuracy Results

- The sampling regression cube improves prediction accuracy significantly
- A regression cube without model reduction is even worse than a single model!



Problem: Traffic Regulator Mapping

Cell phones in vehicles were used as the sources (whose reliability is unknown)

Stopped for 2-10 seconds? → Stop sign

Stopped for 40 seconds – 1 minute? → Traffic light

All reports were fed to a data cleaning/clustering service to determine their probability of correctness

Resulting predictions were compared against ground truth

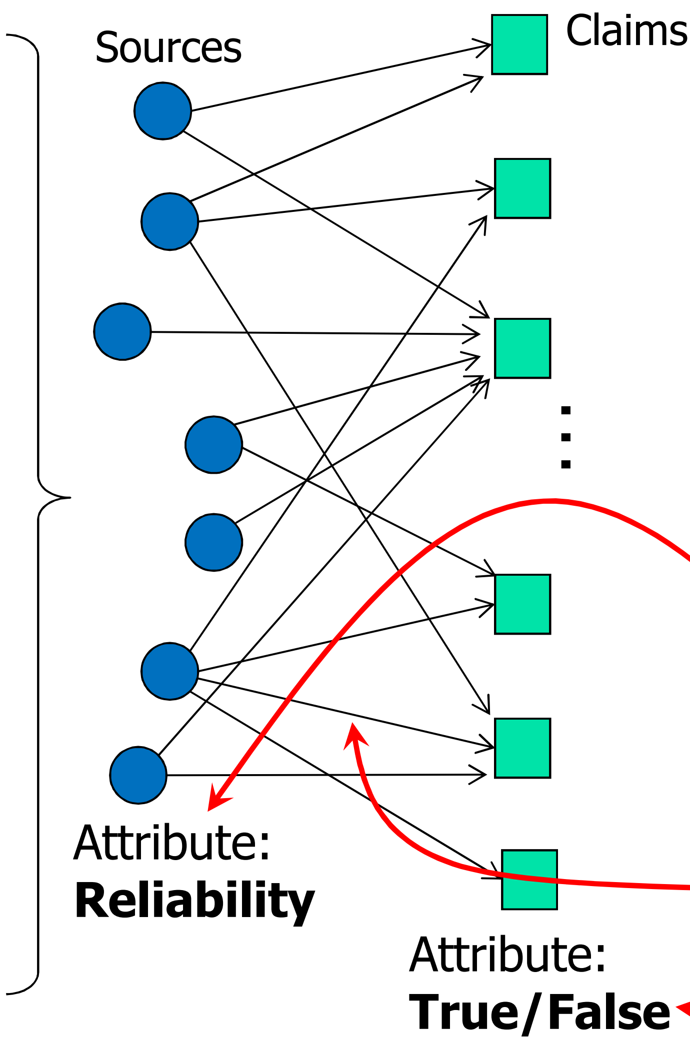
Social Channel "Decoding"
A Maximum Likelihood Estimation Problem

Events

Civil Unrest

Hurricanes

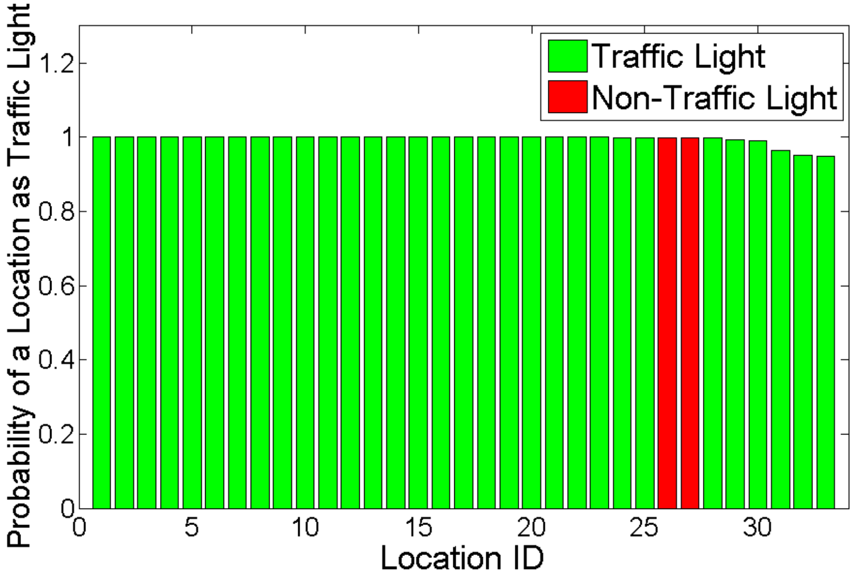
Man-made disasters



- Joint estimation of
  - Source reliability
  - True/false value of each observation
- Given
  - Who said what

$$P(SC|\theta) = \sum_z P(SC, z|\theta)$$

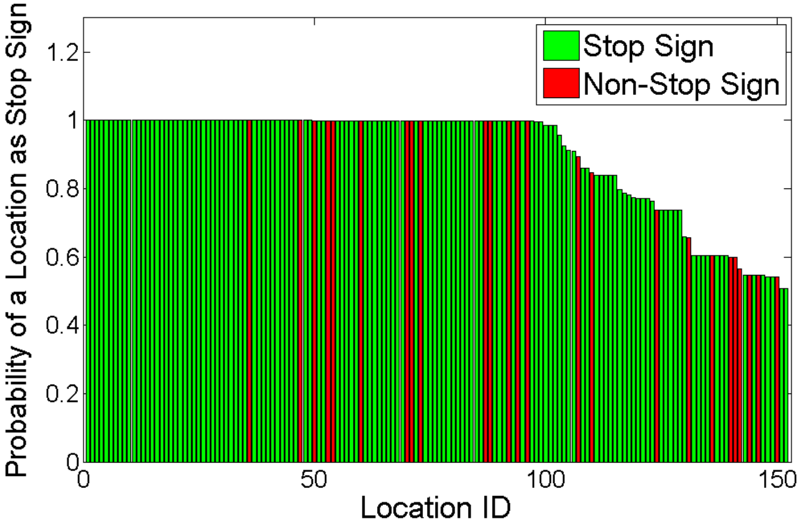
Traffic Regulator Mapping From GPS Data



Traffic Light Location Detection

Experiment setup:
34 drivers, **300** hours of driving in Urbana-Champaign
1,048,572 GPS readings, **4865** claims generated by phone (3033 for stop signs, 1562 for traffic lights)

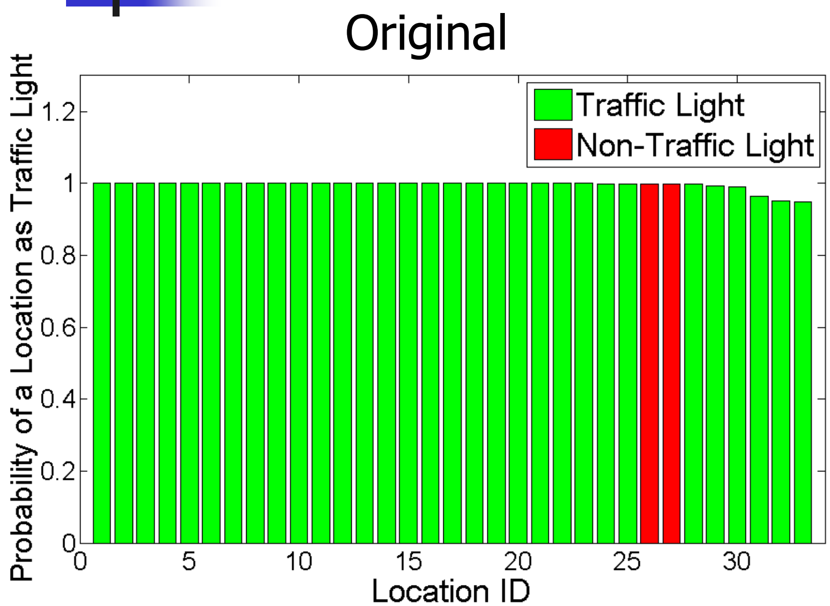
Traffic Regulator Mapping From GPS Data

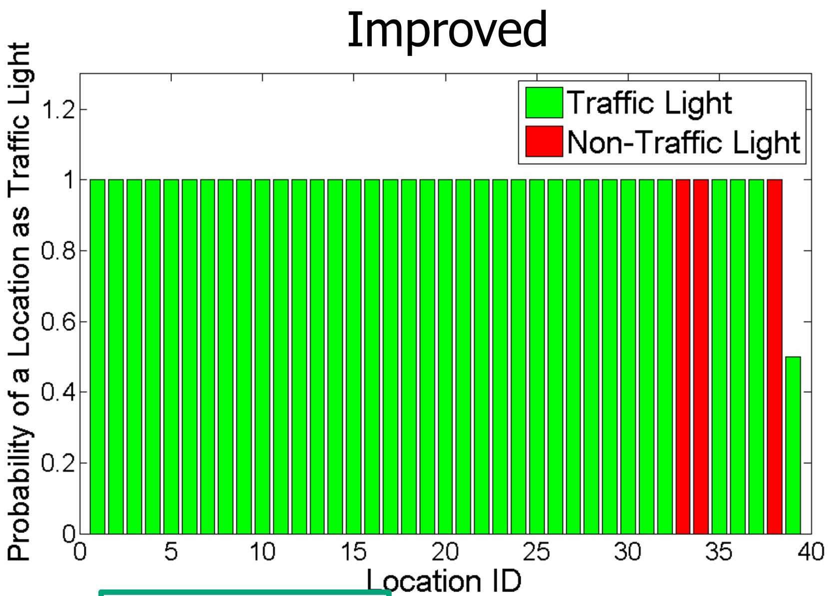


Stop Sign Location Detection

Experiment setup:
34 drivers, **300** hours of driving in Urbana-Champaign
1,048,572 GPS readings, **4865** claims generated by phone (3033 for stop signs, 1562 for traffic lights)

Traffic Regulator Mapping (Enhanced)
Understanding Silence

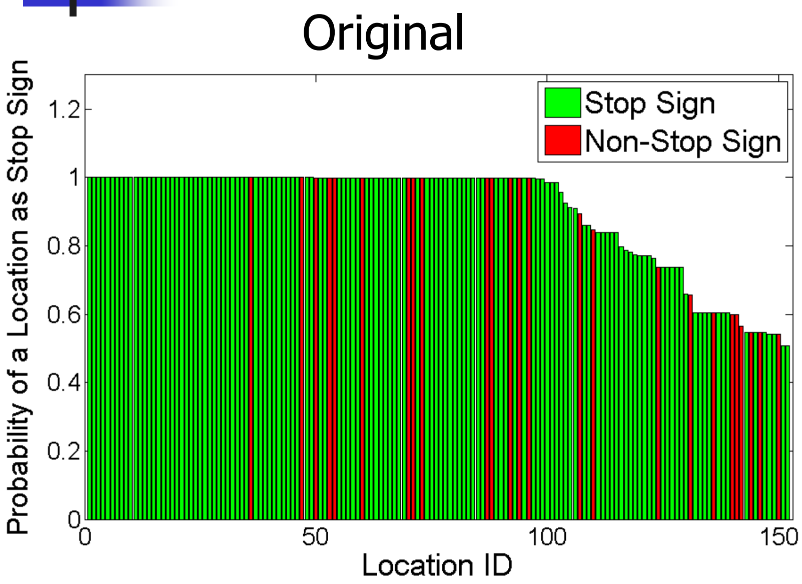


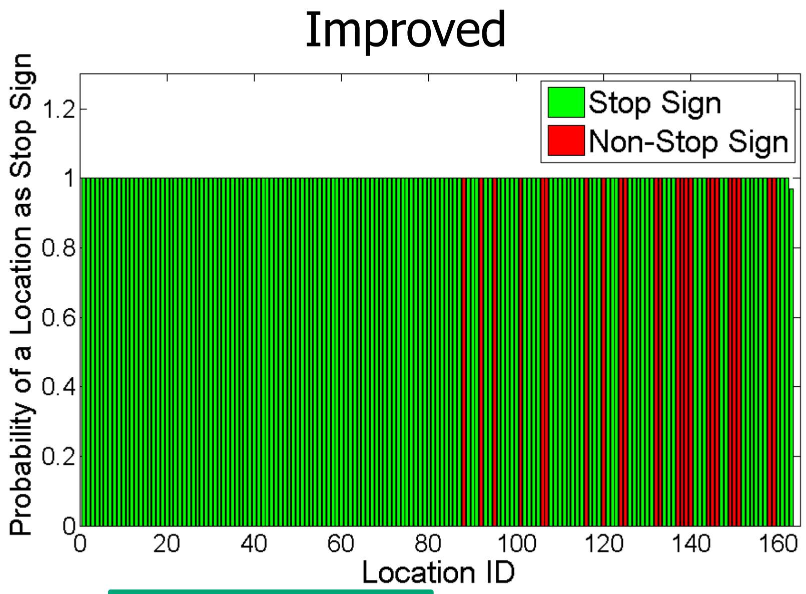


	Original EM	Improved EM
Average Source Reliability Estimation Error	10.19%	**7.74%**
Number of Unbounded Sources	3	**1**
Number of Correctly Identified Traffic Lights	31	**36**
Number of Mis-Identified Traffic Lights	**2**	3

Traffic Regulator Mapping (Enhanced)

Understanding Silence

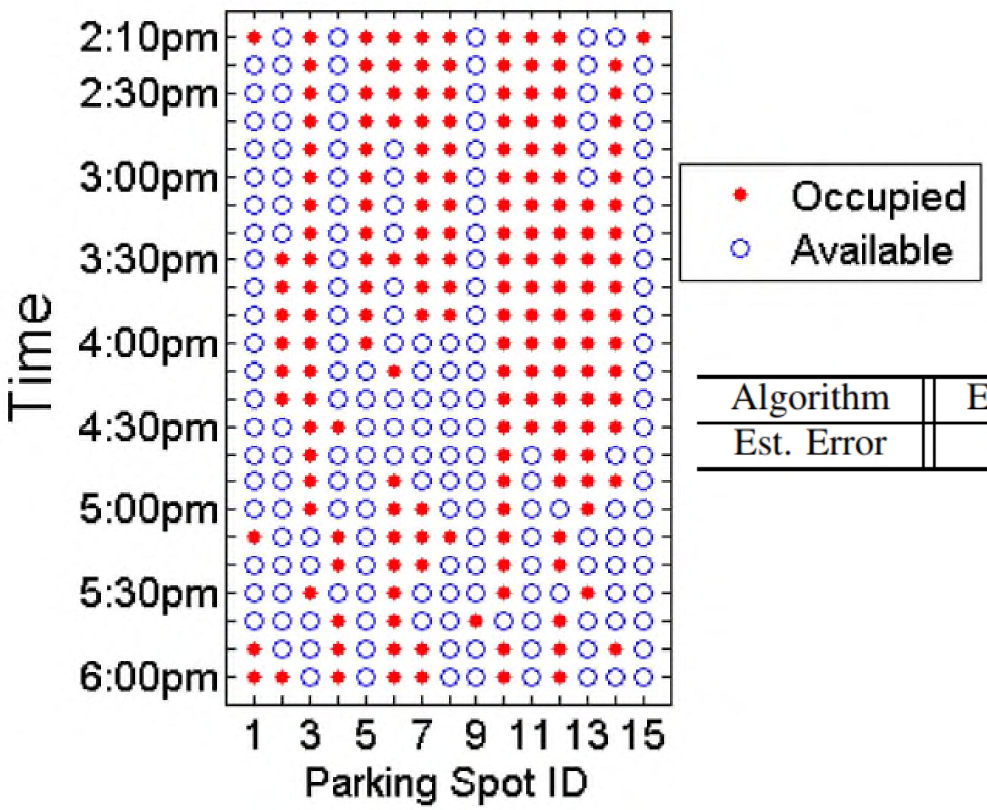




	Original EM	Improved EM
Average Source Reliability Estimation Error	20.06%	**14.32%**
Number of Unbounded Sources	5	**1**
Number of Correctly Identified Traffic Lights	127	**139**
Number of Mis-Identified Traffic Lights	25	**24**

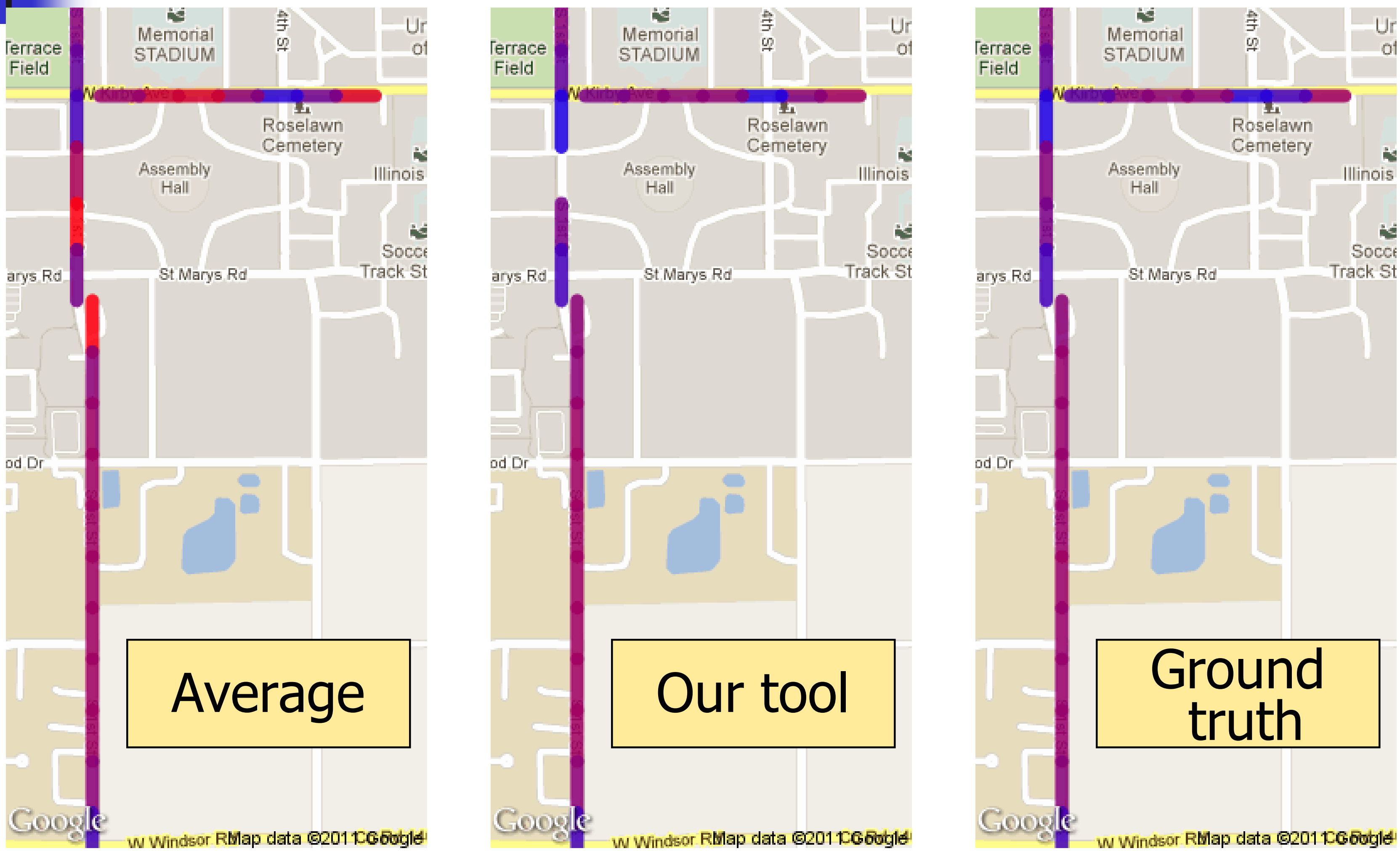
Example with Time-varying Ground Truth State

- Estimating empty parking spots from unreliable observers



Algorithm	EM-VTC	EM-C1	EM-R1	Voting	History
Est. Error	**0.0319**	0.1855	0.2261	0.2203	0.1420

Problem:
Cleaning Noisy Speed Data



The Age of Social Broadcast

The Past

O(n)

O(n)

The Age of Social Broadcast

$O(n)$

$O(n)$

$O(n)$

$O(n^2)$

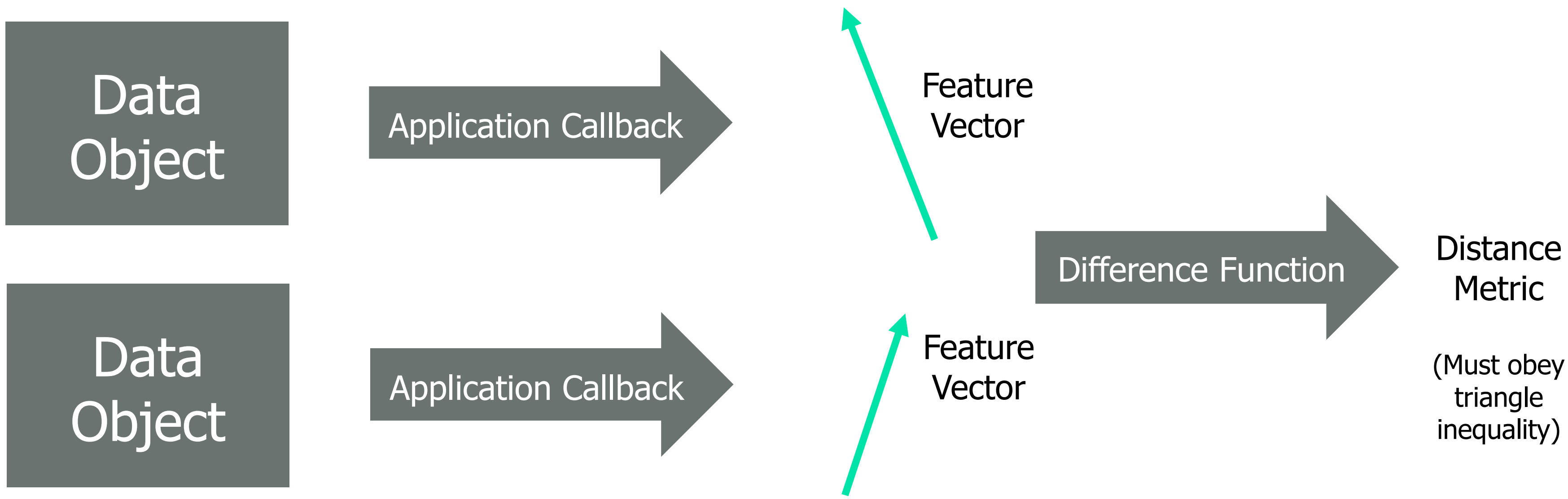
$O(n) \rightarrow O(n^2)$

The Present

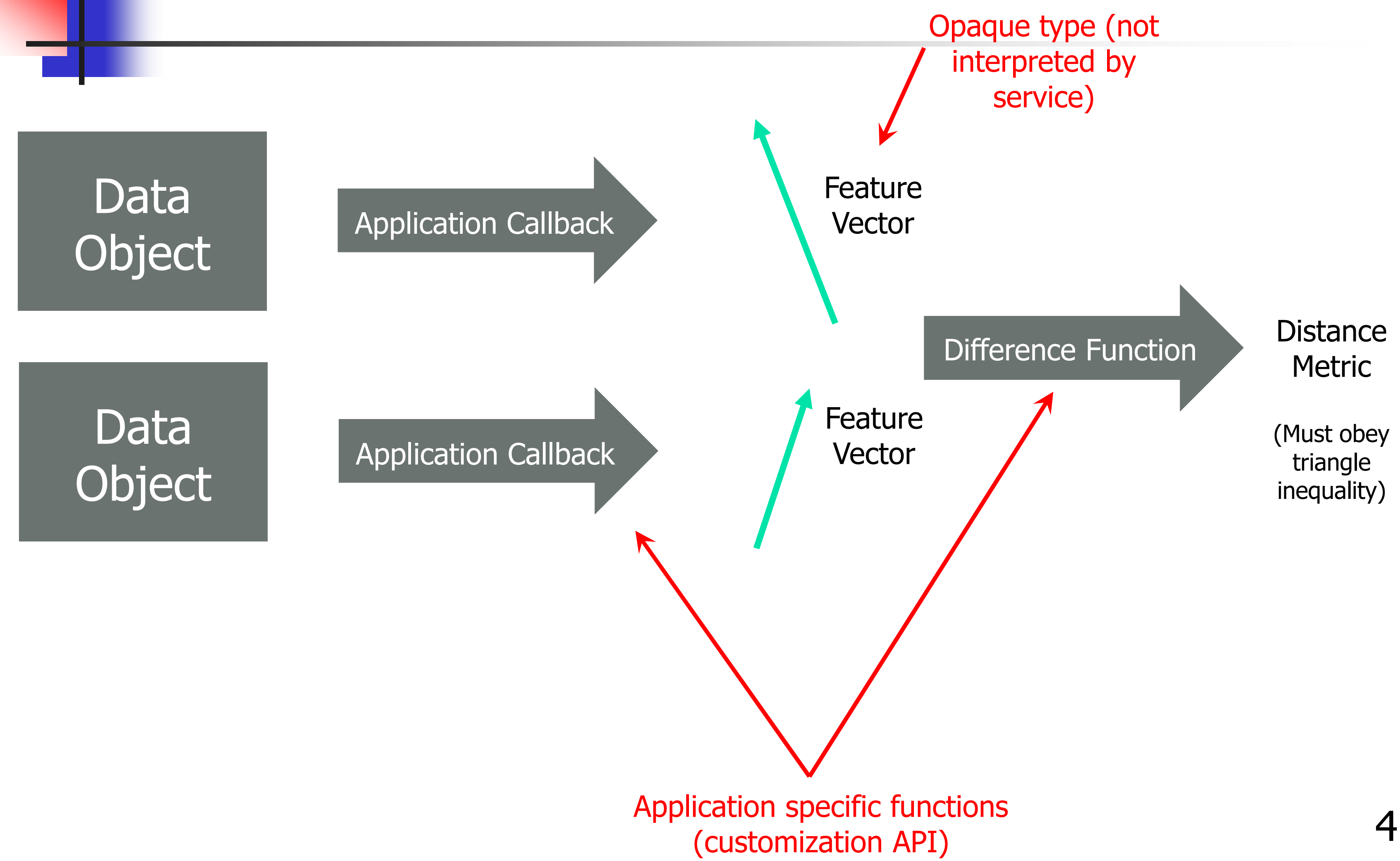
Challenge: Extractive Summarization

Build a data service that allows applications to retrieve (extractive) data summaries at arbitrary levels of granularity in accordance with an application-specific redundancy metric

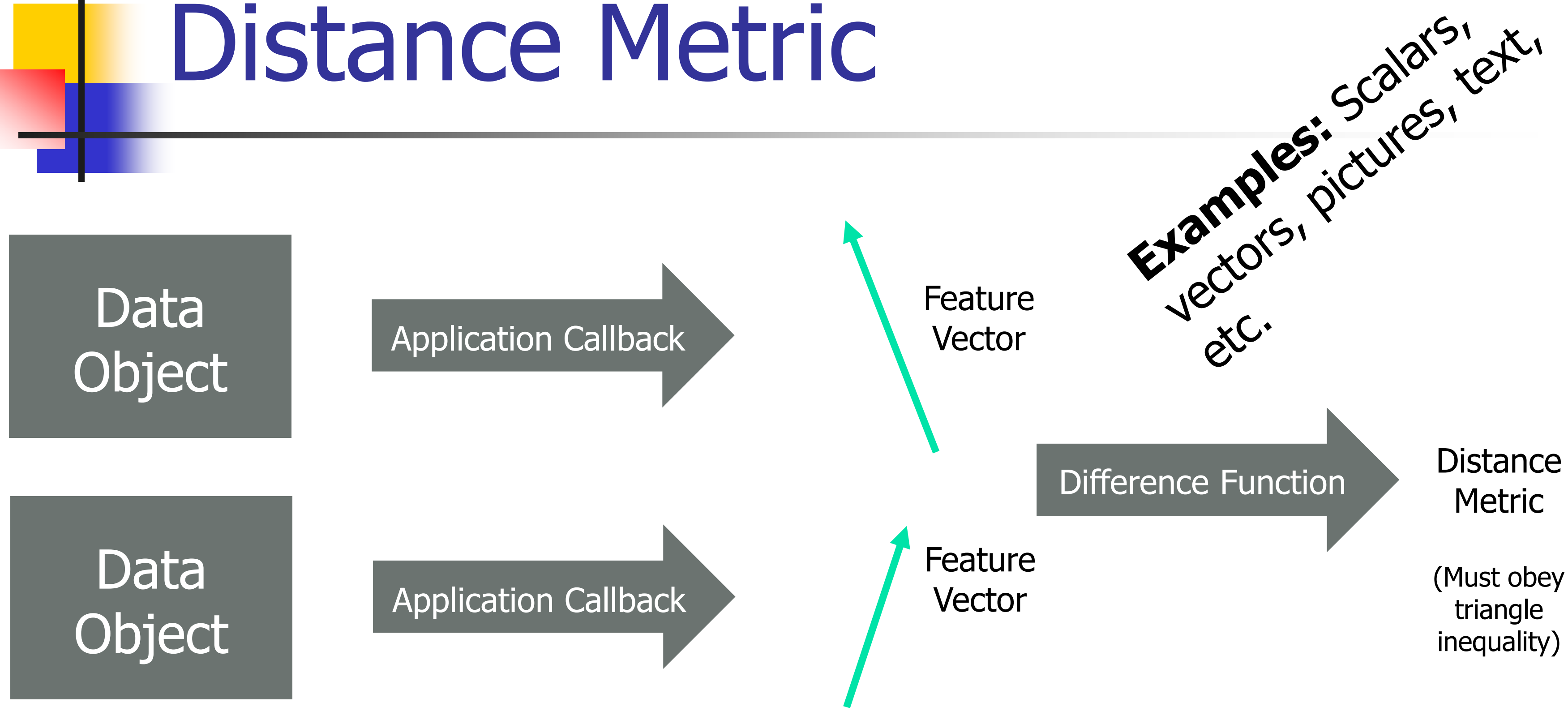
Customizability: The Distance Metric



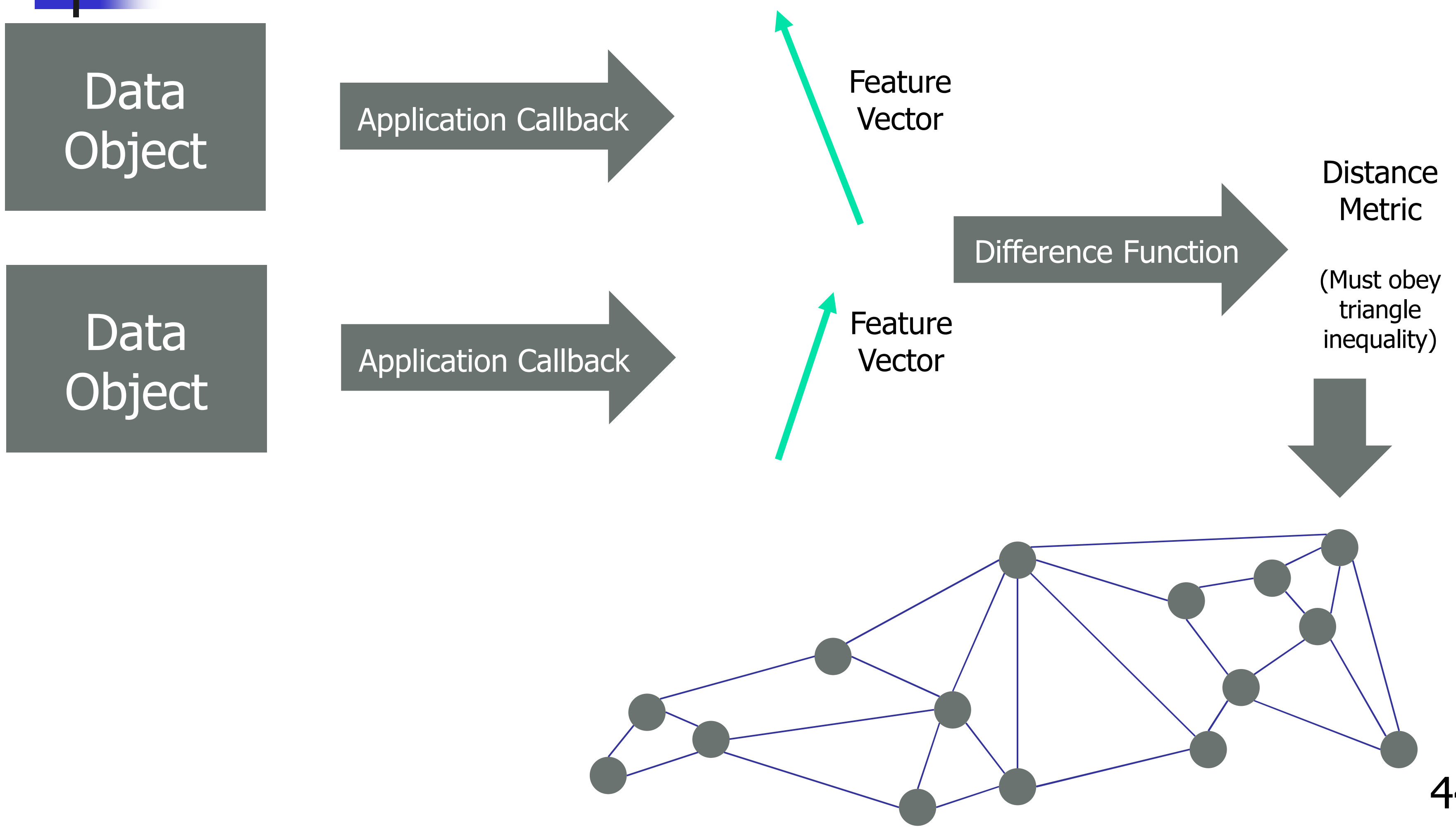
Customizability: The Distance Metric



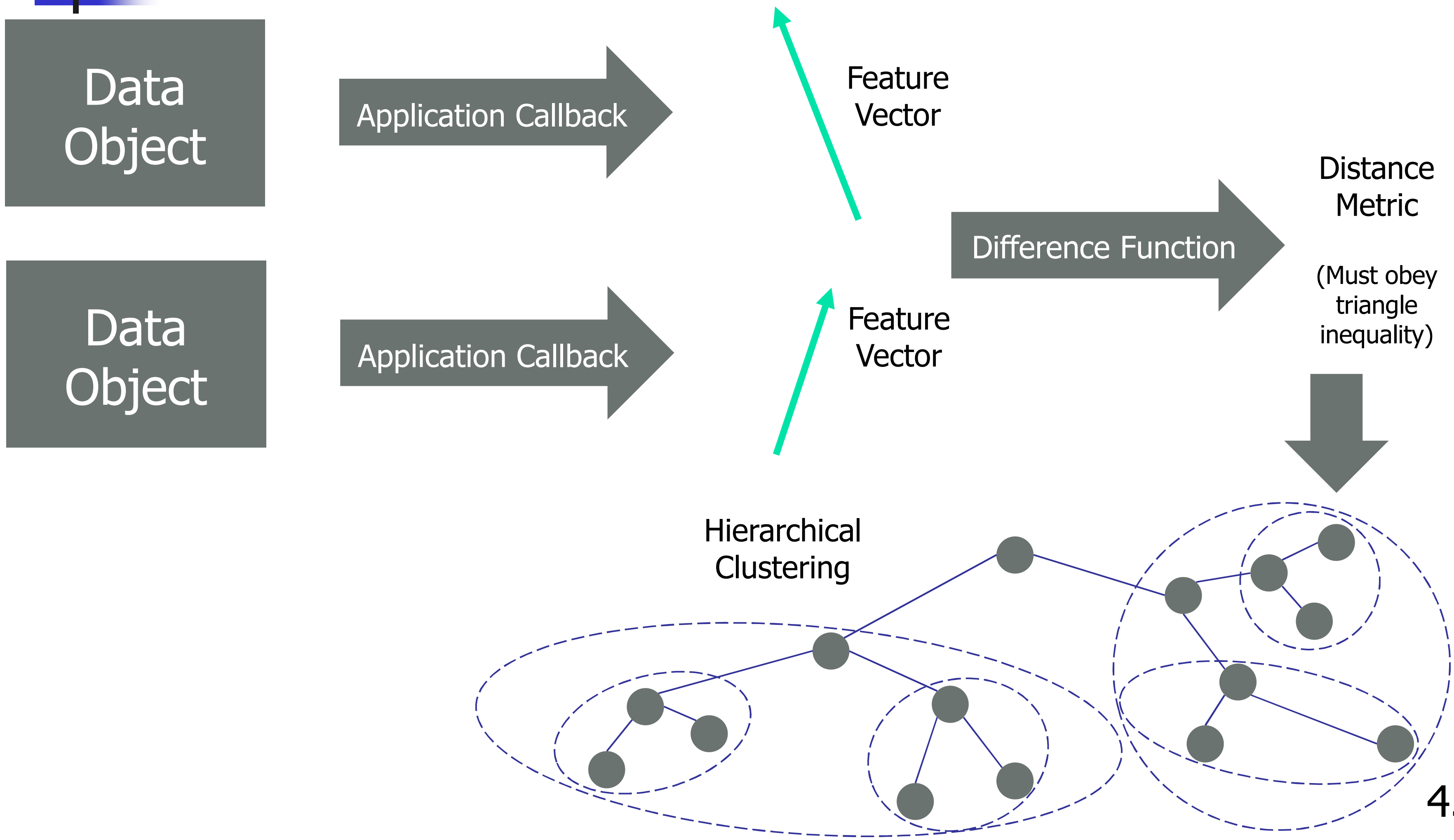
Customizability: The Distance Metric



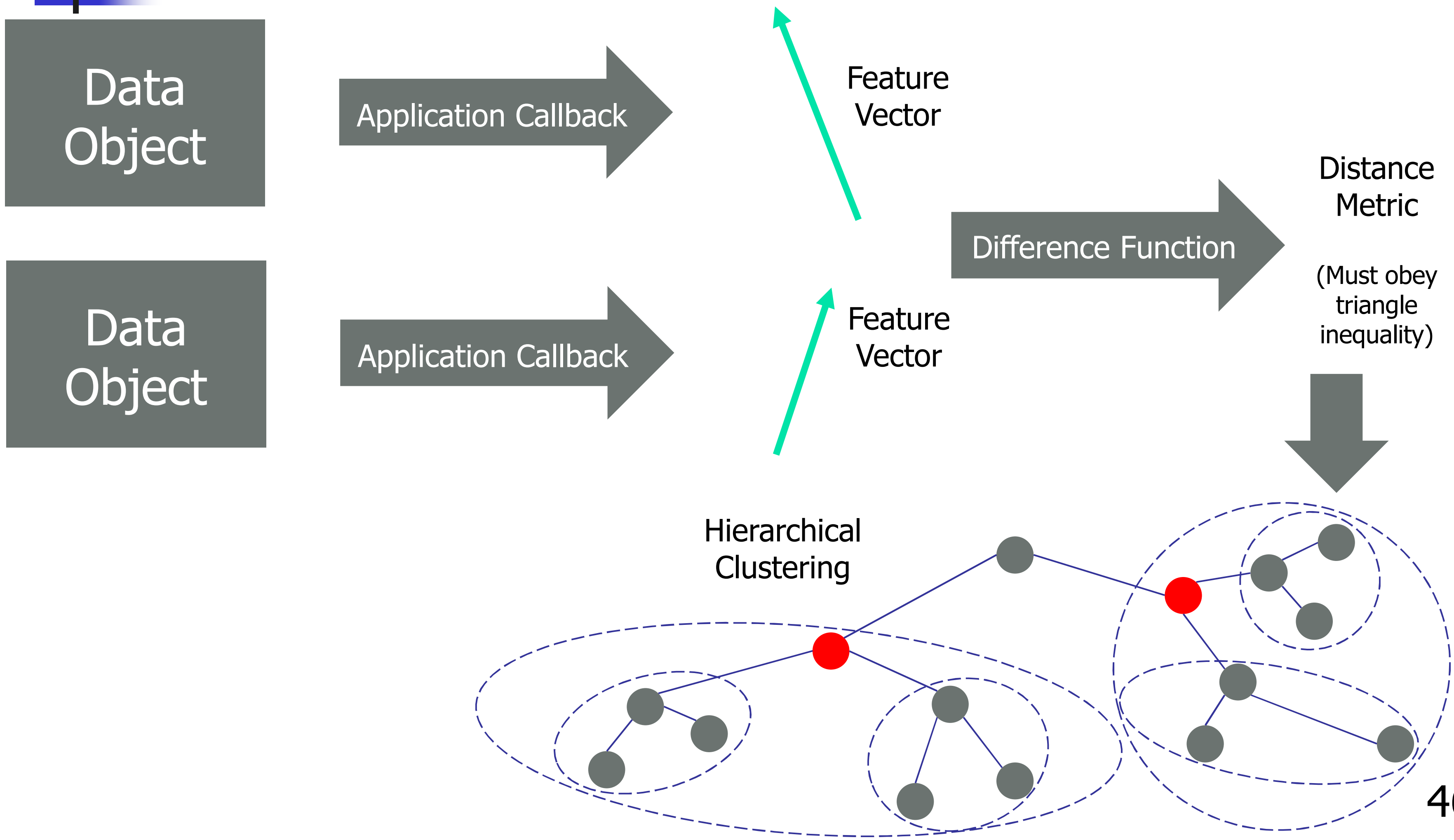
Customizability: The Distance Metric



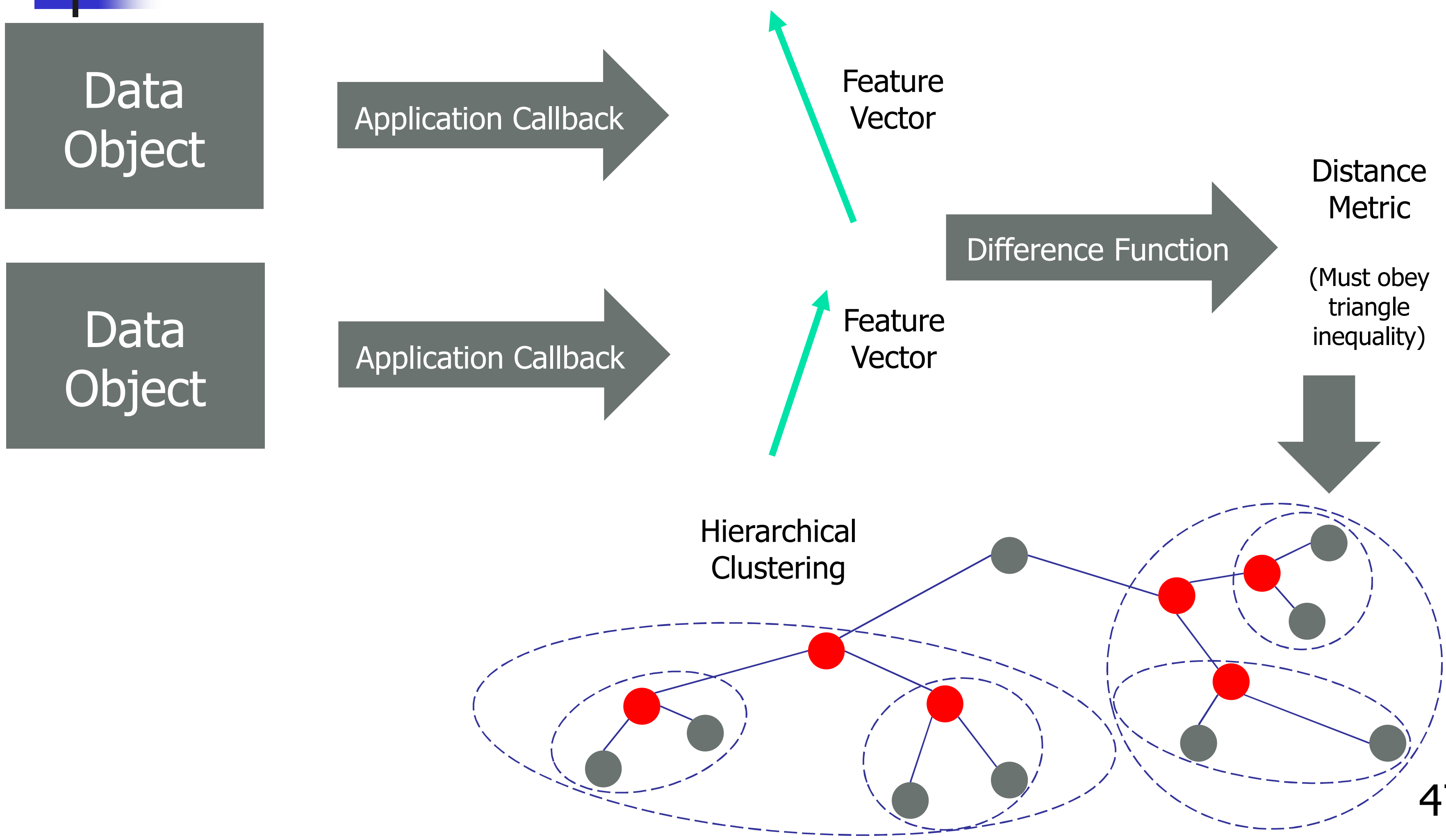
Customizability: The Distance Metric



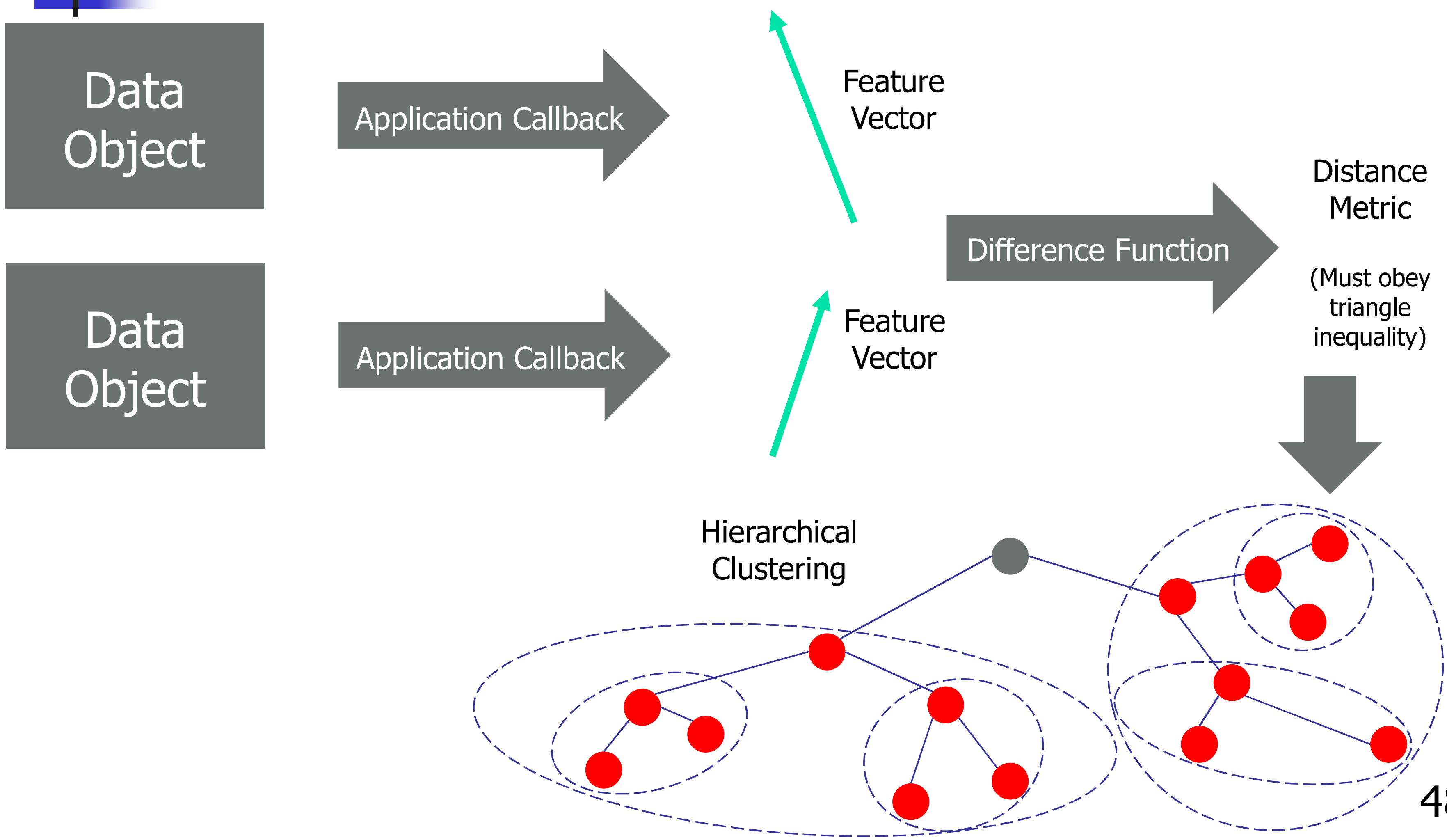
Summarization



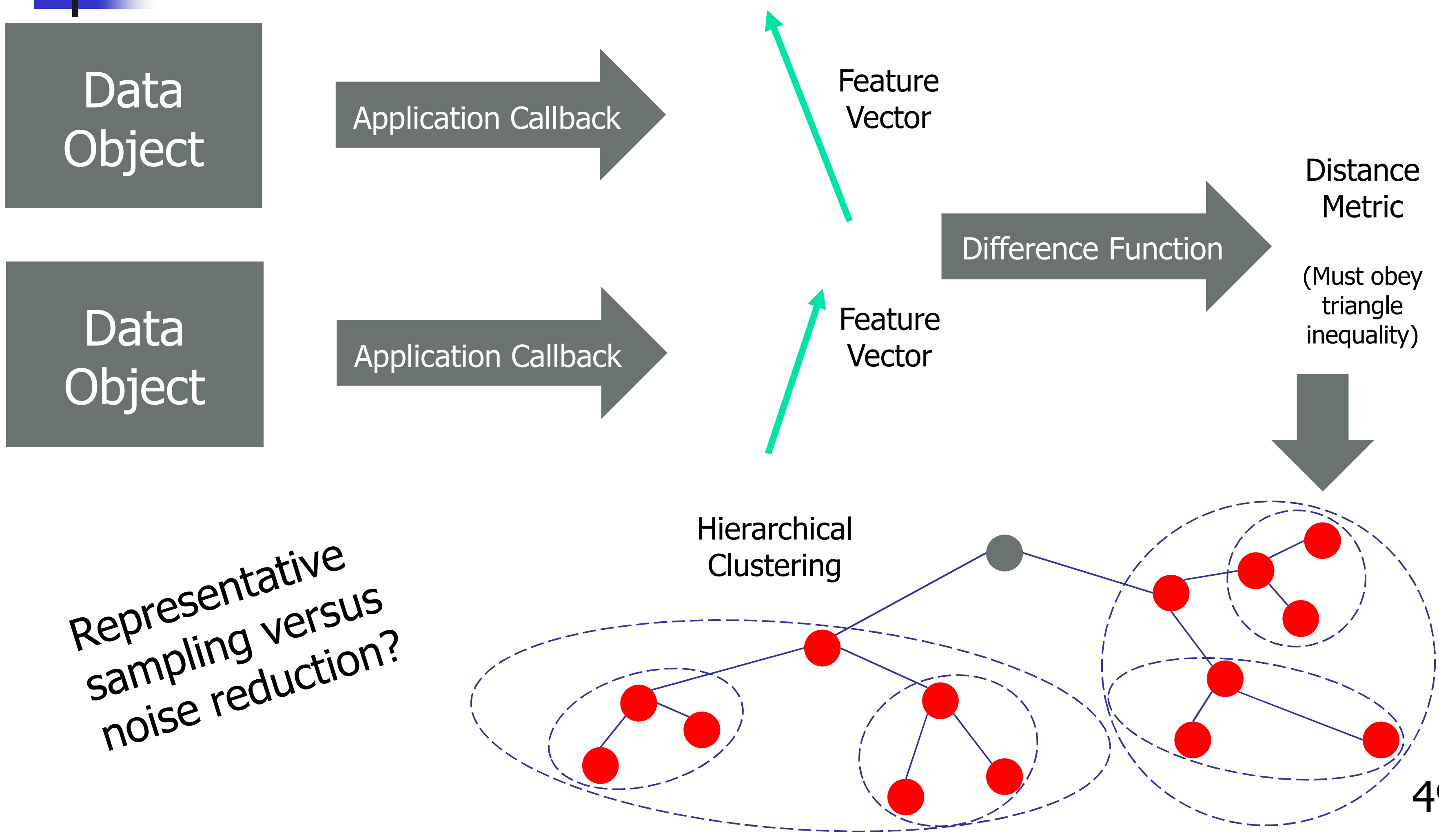
Summarization



Summarization



Summarization



A Network Paradigm Shift

Communication → Information Distillation

The data fire-hose effect

- Present Networks

Goal:

Communication

- Maximizes bit throughput between end-points
- Most data is "logical"
- Protocols geared primarily for point-to-point communication
- Data loss may be a problem

- Future Distillation Networks

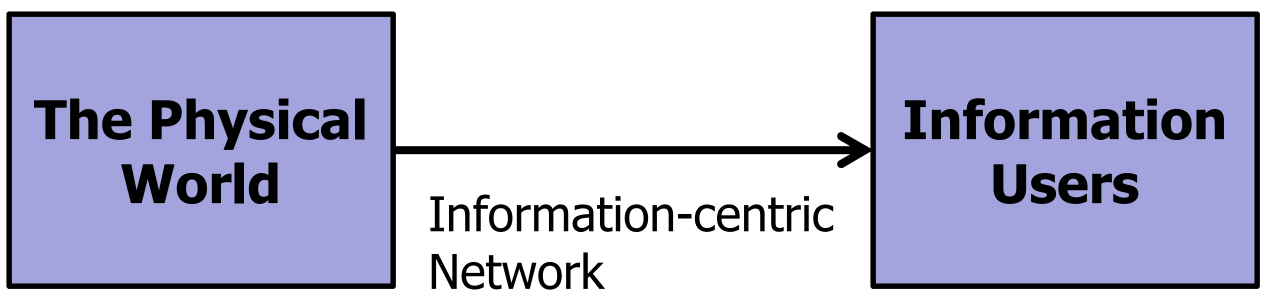
Goal:

Information Distillation

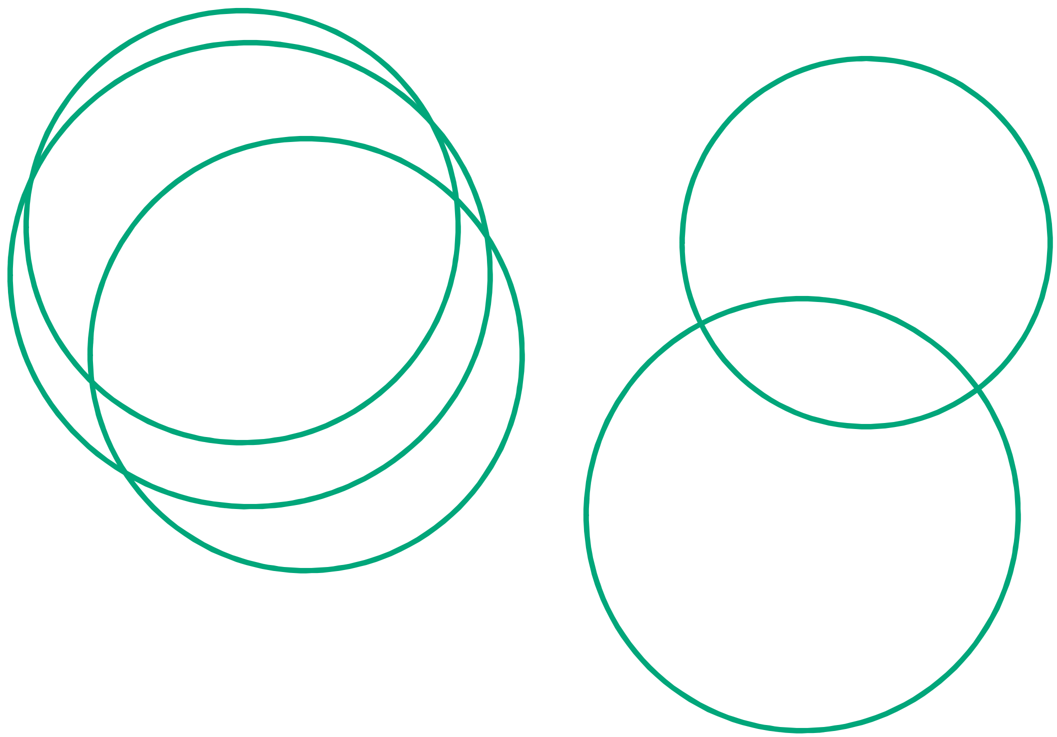
- Maximizes *information flow*
- Much data is "physical"
- Protocols geared for data filtering, and aggregation
- Data loss may be a feature intended to reduce less informative bits

A Primary Network Design Challenge

- How to build networks that *maximize useful information flow from the physical world*?

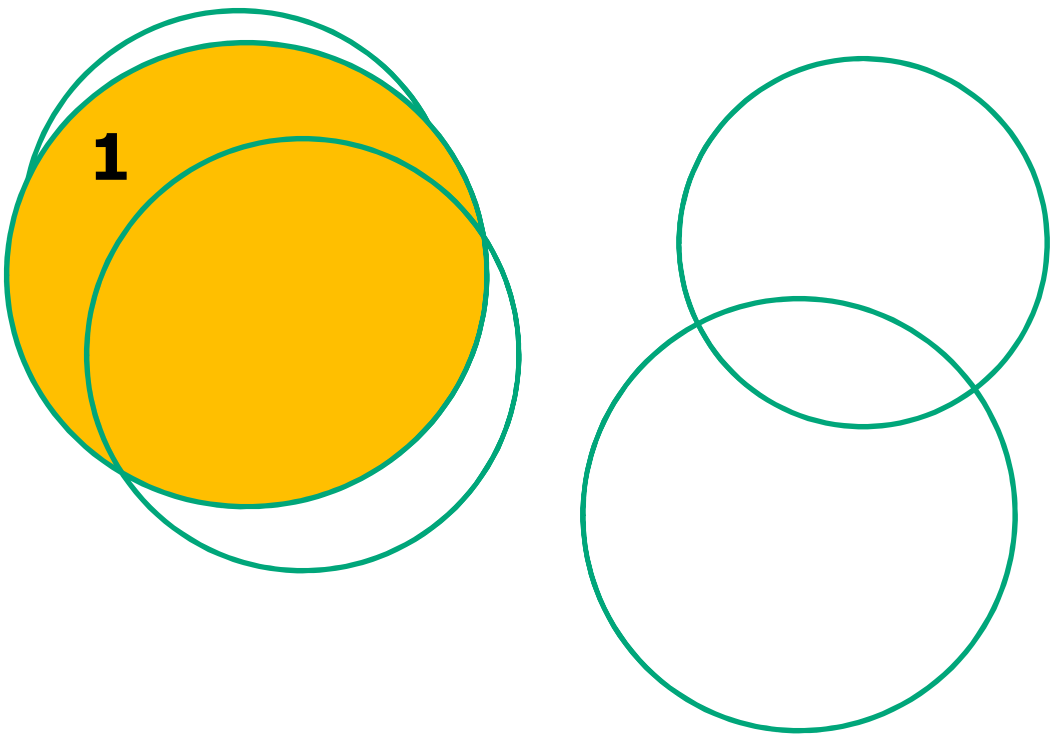


Information-maximizing Prioritization



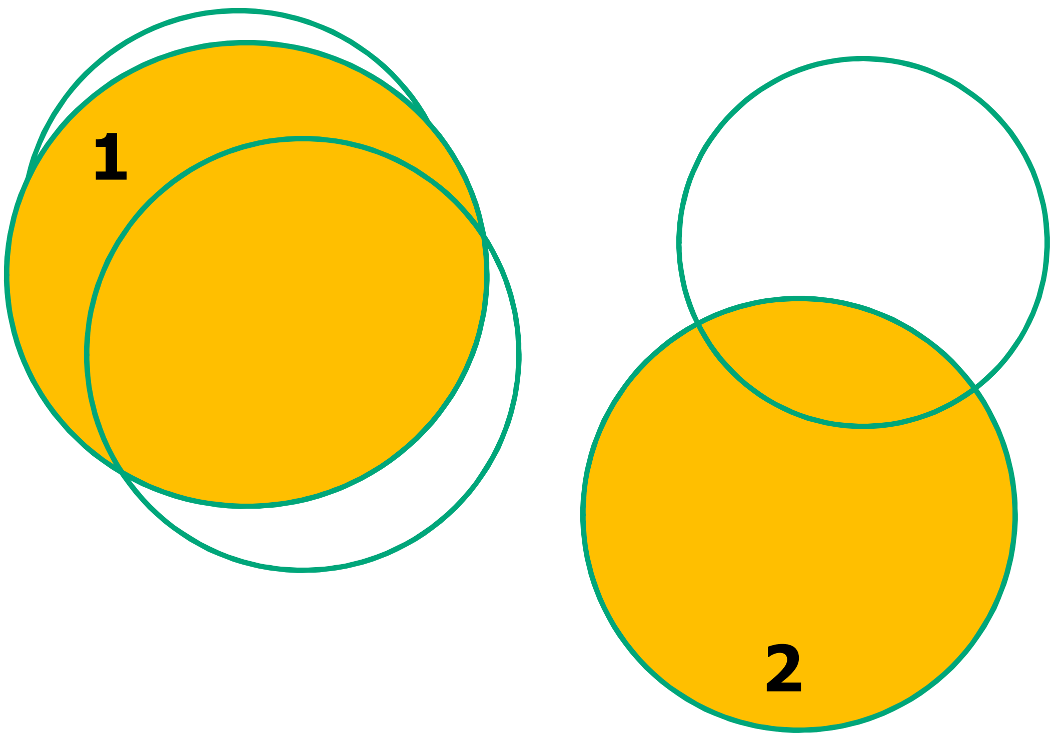
- Determine transmission order?

Information-maximizing Prioritization



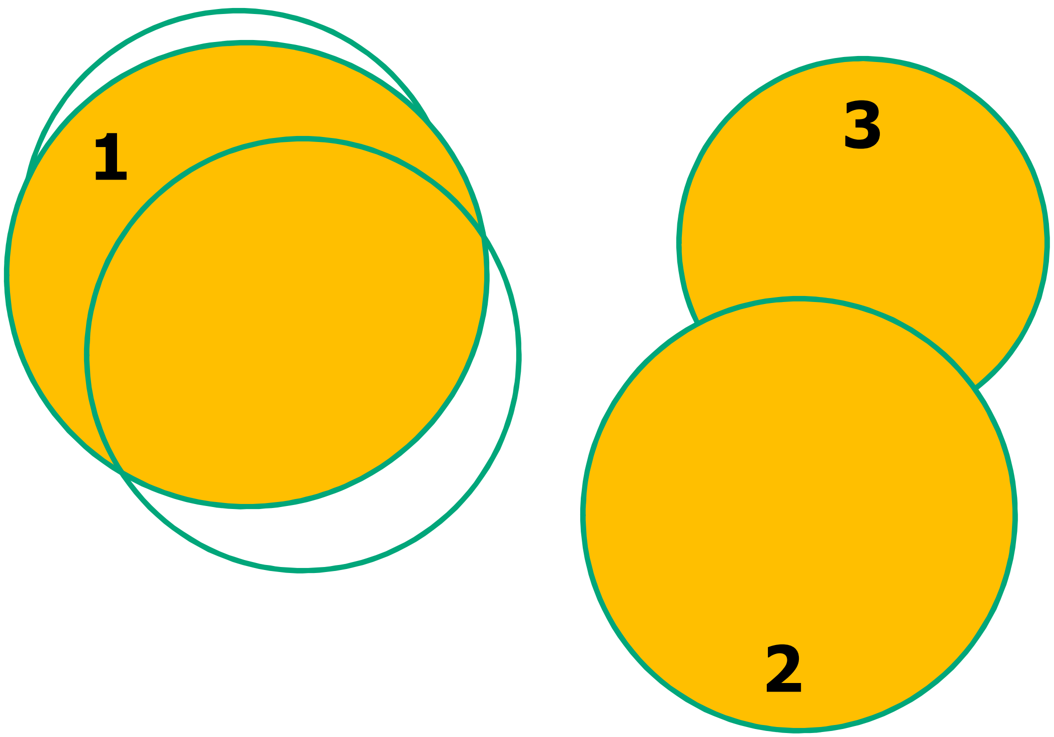
- Determine transmission order?

Information-maximizing Prioritization



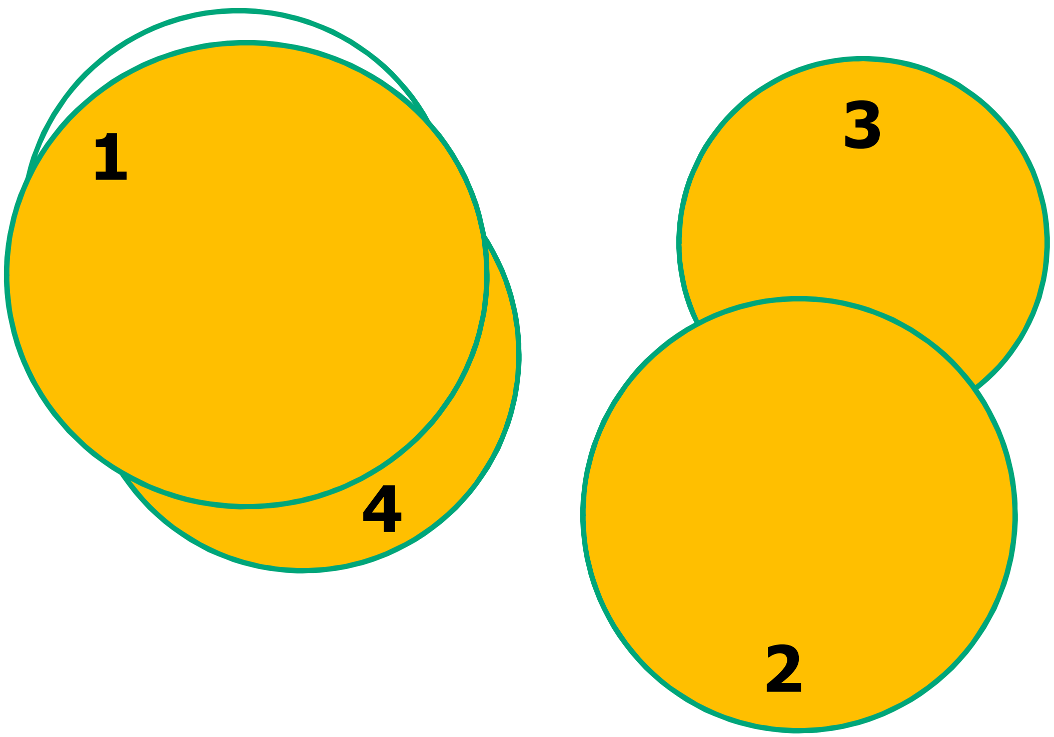
- Determine transmission order?

Information-maximizing Prioritization



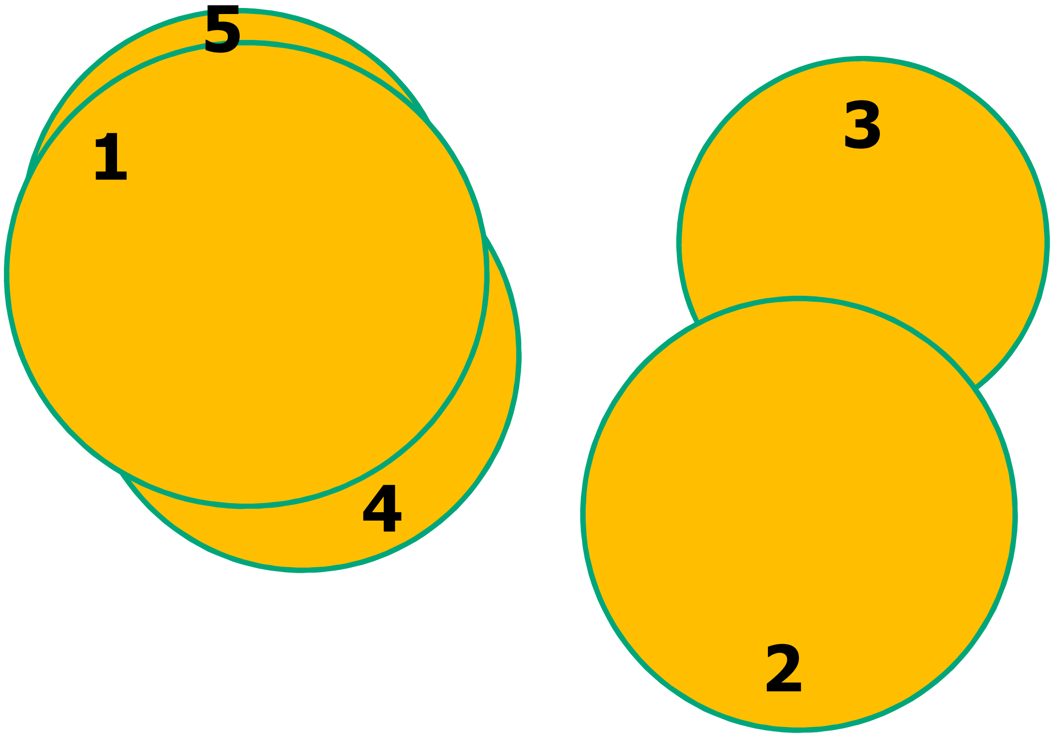
- Determine transmission order?

Information-maximizing Prioritization



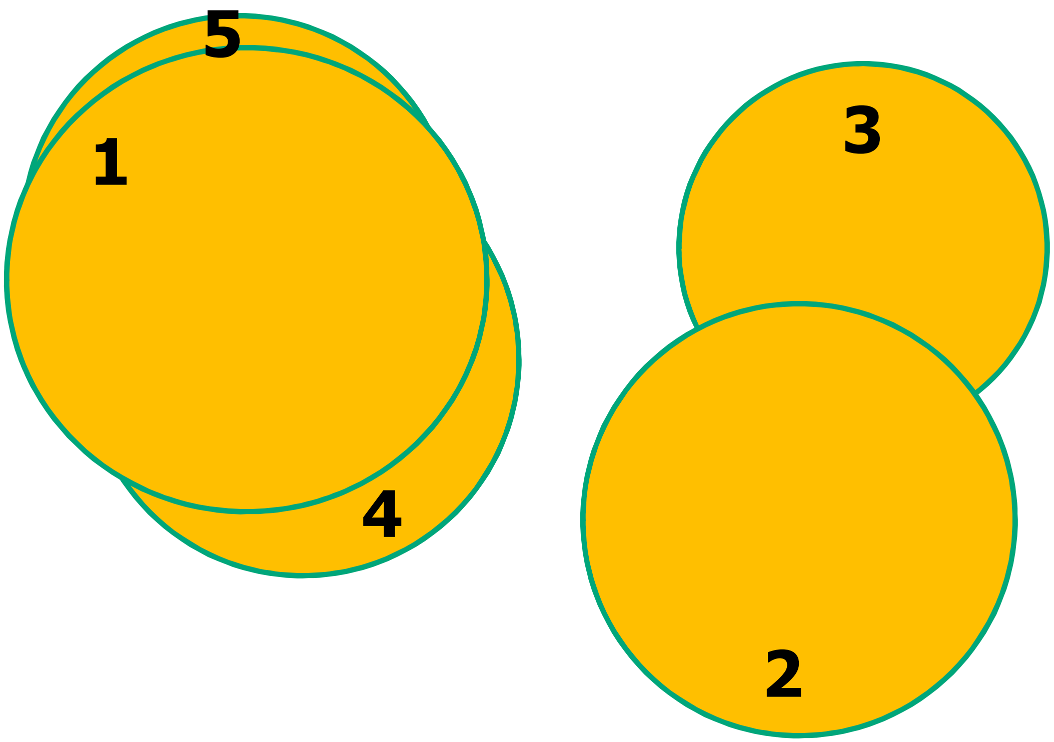
- Determine transmission order?

Information-maximizing Prioritization



- Determine transmission order?

Information-maximizing Prioritization

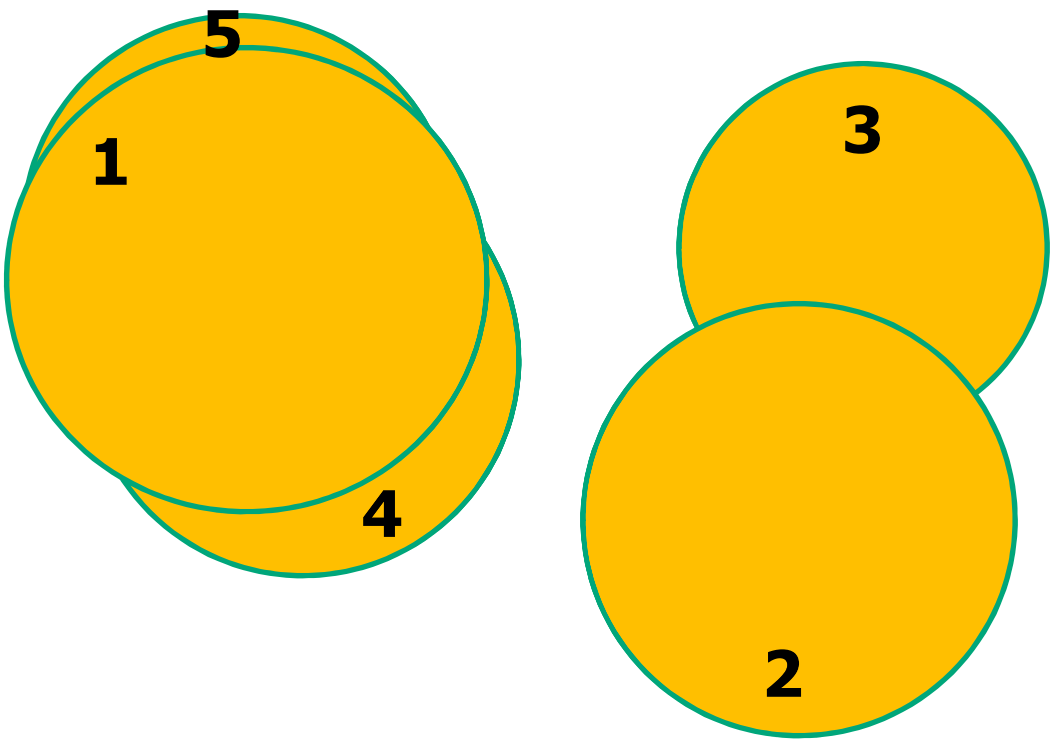


- Determine transmission order?

Coverage-monotonic scheduling

Information-maximizing Prioritization

Note: Coverage can be defined in an abstract feature space



Coverage-monotonic scheduling

A Disaster Response Scenario

- A big disaster strikes a city…

Images are collected from the Internet

Hurricane Katrina 2005

Nepal earthquake 2015

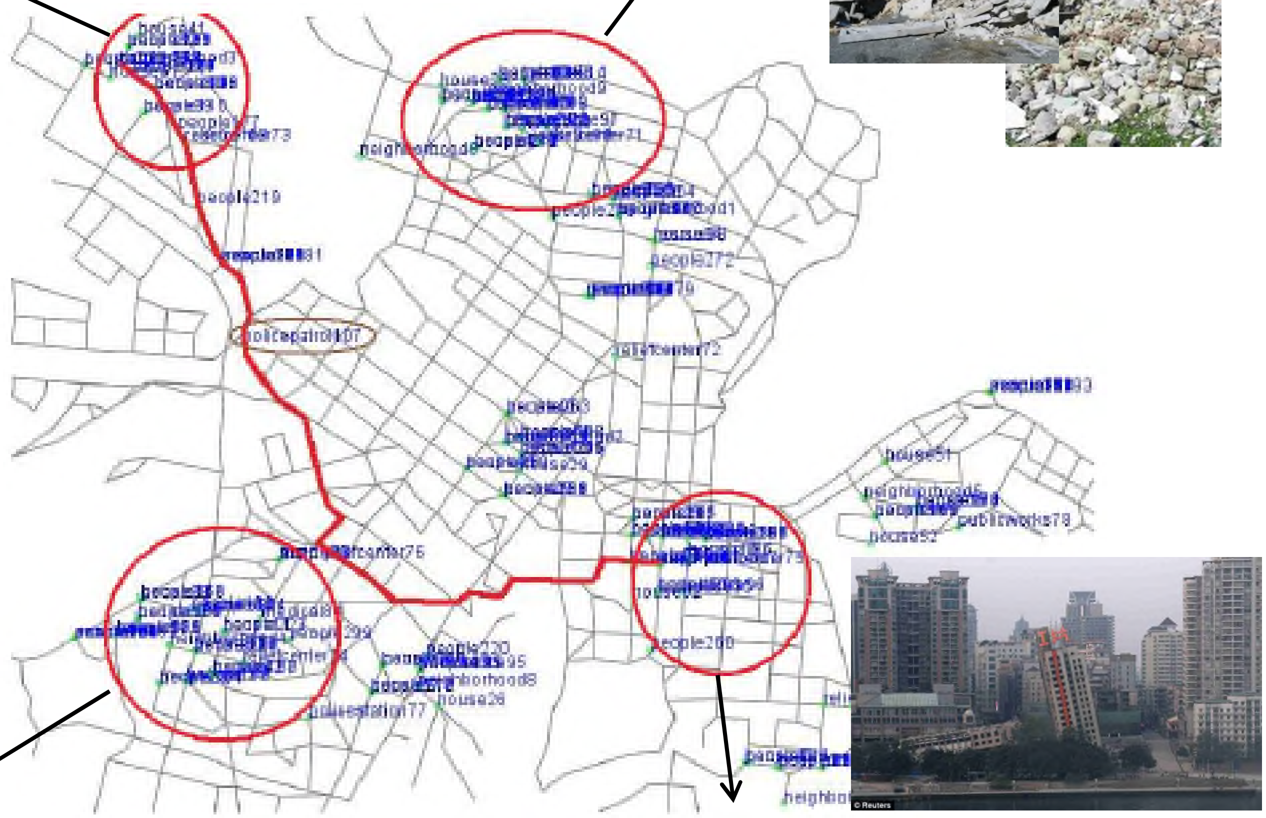
Thailand flood 2011

- Volunteers are recruited: They scout the area, capture pictures and send them to a rescue center
- Network constraints prevent sending all pictures

Problem: Data Selection to Maximize Coverage

Fire on 6th and Main.

Collapse on Park Ave.



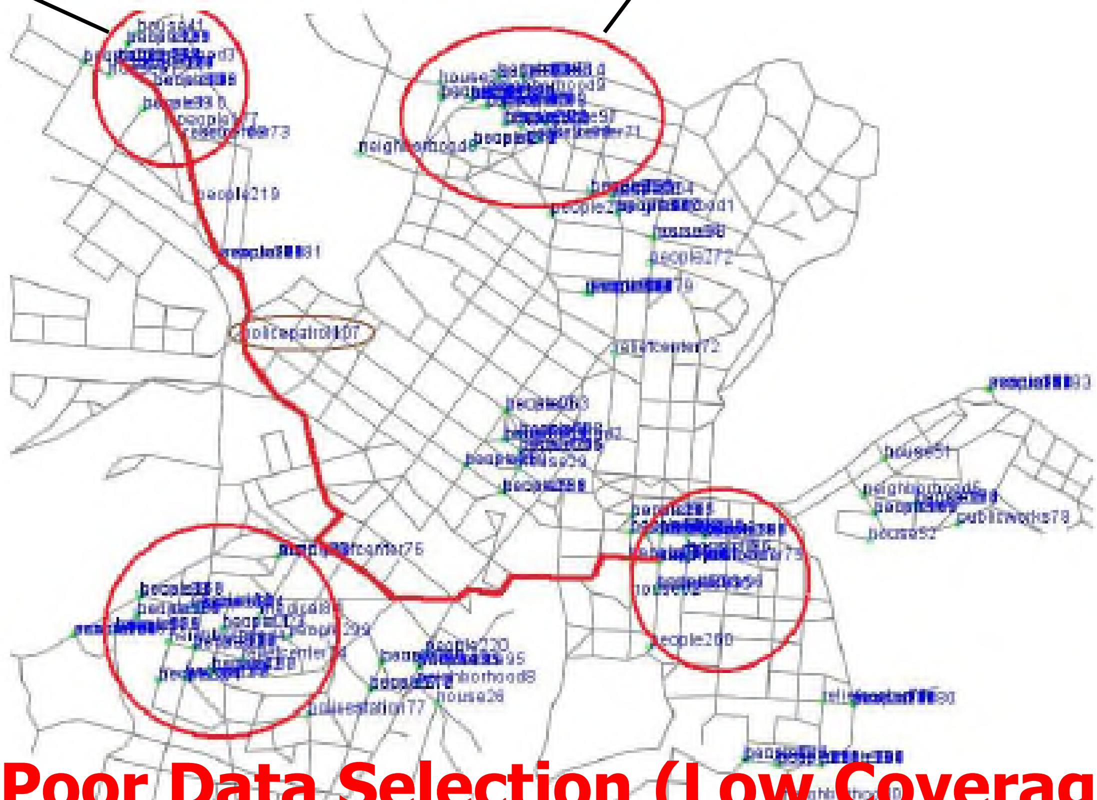
Flooding on State St.

Structural damage on Pier Square

Example of Bad Coverage

Fire on 6th and Main.

Collapse on Park Ave.

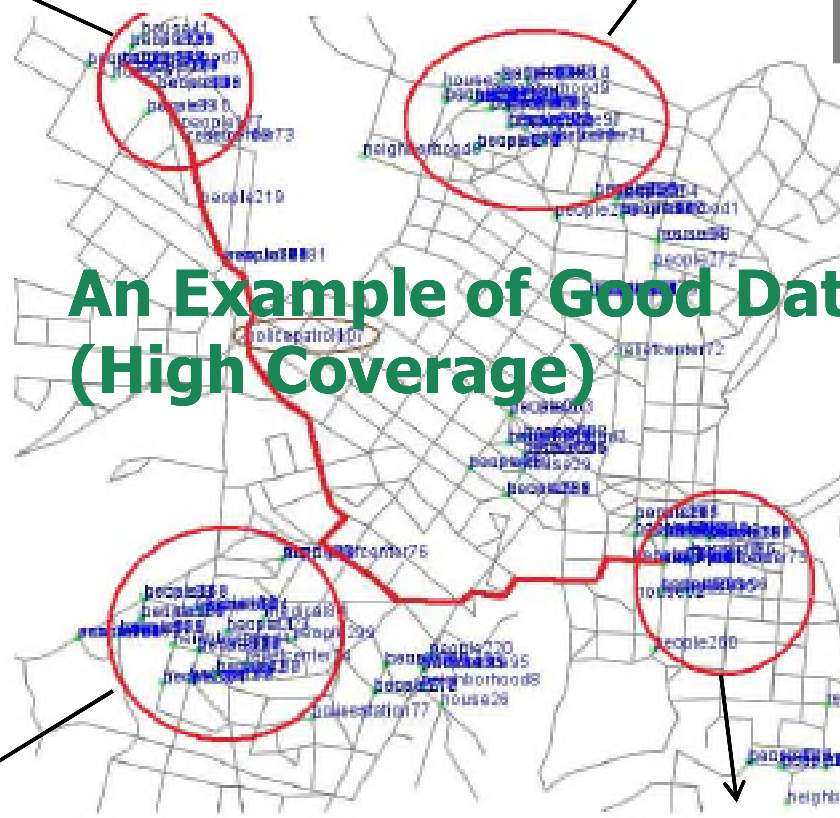


An Example of Poor Data Selection (Low Coverage)

Example of Good Coverage

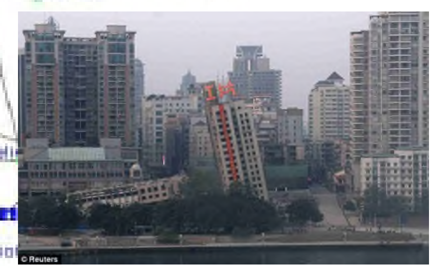
Fire on 6th and Main.

Collapse on Park Ave.



An Example of Good Data Selection (High Coverage)

Flooding on State St.



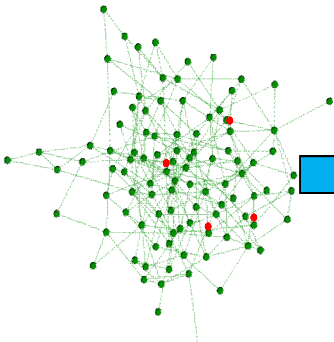
Structural damage on Pier Square

A Scheduling Approach: Coverage-maximizing Priorities

- Implement coverage-maximizing in-network prioritization for forwarding and storage
  - Objects are forwarded/dropped in a priority order aimed to maximize coverage of delivered content
    - Objects similar to previously forwarded ones get lower priority
  - Challenge: Forwarding and dropping must be made aware of the degree of semantic redundancy (i.e., similarity) between objects

Project Ideas: Robustness in Human-centric CPS

Robustness of Resources



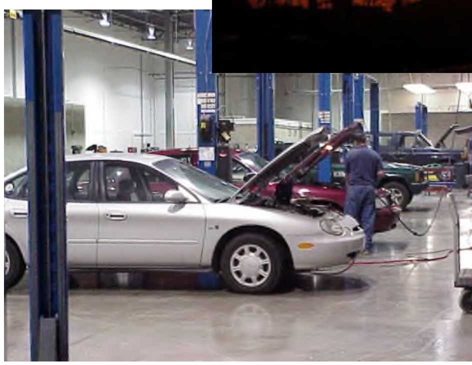
Robustness of Data Access

Robustness of Information Extraction

Robustness of Inference

- ***Ensure Robustness of:***
  - ***Underlying physical resources:*** A set of inter-dependent resource networks (e.g., for data transport, power, and physical mobility)
  - ***Data communication and storage resources:*** A digital plane that offers routing, storage, and capacity to access raw data
  - ***Information resources:*** Information filters for assessing quality of information and for filtering higher-quality information from raw data
  - ***Inference processes:*** Tools for modeling, estimation, and prediction of latent variables relevant to decision support

Failures in Complex Systems



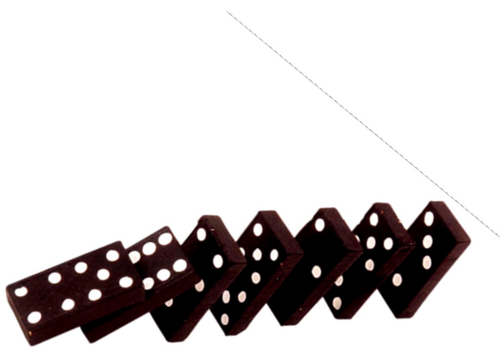
When systems fail, a common goal is:

Localize and fix the root cause!

Complexity Reduction: Simplifying Dependencies

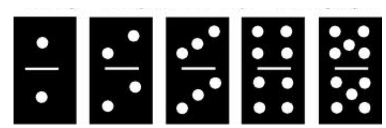
Reduce interactions and coupling

- Reduces propagation of local failures globally



Tightly coupled

Robustness solutions



Less coupled

The Performance/Robustness Trade-off

Performance: Exploring the edge of stability with global knowledge (global → more dependencies)

Robustness: Guaranteeing delivery in the face of adverse conditions and limited knowledge

Interactive Complexity in Cyber-Physical Systems

- Performance optimizations lead to:

  → Complex interactions (e.g., global versus local)

  → More dependencies

  → Deeper cascading failures

  → Lower robustness

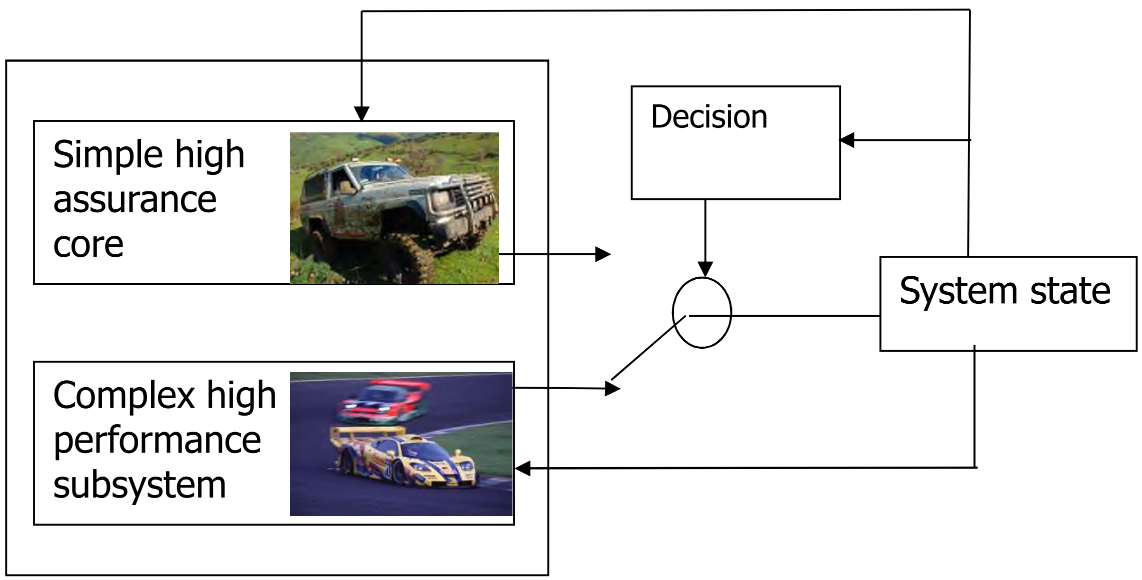
Cascading failure on "high-performance" road

Non-cascading failure on side-street

Achieve both Performance and Robustness *together* ?

The Simplex Architecture (by Lui Sha)

A simple verifiable core; diversity in the form of 2 alternatives; feedback control of the software execution.



Data Flow Block Diagram