

CZ4079 Final Year Project

A Machine Learning-Based Approach to Time-Dependent Shortest Path Queries

Wei Yumou

School of Computer Science and Engineering
Nanyang Technological University



Agenda

- 1 Introduction
- 2 Preliminary Processing
- 3 Landmark Graph
- 4 Travel Time Estimation



Introduction: Problem



Introduction: Problem

- A **dynamic road network** $G = (V, E)$ with a time-dependent weight function $w : E, t \rightarrow \mathbb{R}$

Introduction: Problem

- A **dynamic road network** $G = (V, E)$ with a time-dependent weight function $w : E, t \rightarrow \mathbb{R}$
- A **query** $Q(u, v, t)$ that asks for a shortest path from u to v departing at time moment t

Introduction: General Approach



Introduction: General Approach

- Traditional **Bellman-Ford or Dijkstra's algorithm** do not work with dynamic edge weights (“the curse of traditionality”)

Introduction: General Approach

- Traditional **Bellman-Ford or Dijkstra's algorithm** do not work with dynamic edge weights (“the curse of traditionality”)
- The new **machine learning-based approach** draws on collective wisdom of thousands of taxi drivers

Introduction: General Approach

- Traditional **Bellman-Ford or Dijkstra's algorithm** do not work with dynamic edge weights (“the curse of traditionality”)
- The new **machine learning-based approach** draws on collective wisdom of thousands of taxi drivers
- **Unsupervised learning** is employed to figure out the time-dependent edge costs

Introduction: General Approach

- Traditional **Bellman-Ford or Dijkstra's algorithm** do not work with dynamic edge weights (“the curse of traditionality”)
- The new **machine learning-based approach** draws on collective wisdom of thousands of taxi drivers
- **Unsupervised learning** is employed to figure out the time-dependent edge costs
- A modified Dijkstra's algorithm calculates a shortest path on the fly

Introduction: Challenges

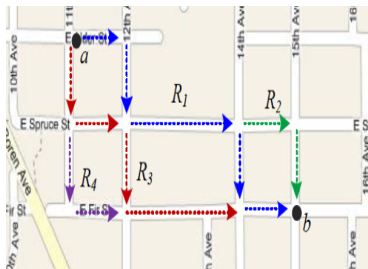


Introduction: Challenges

- Arbitrary u and v

Introduction: Challenges

- Arbitrary u and v
- Sparse sample points



Introduction: Challenges

- Arbitrary u and v
- Sparse sample points
- Limited GPS accuracy

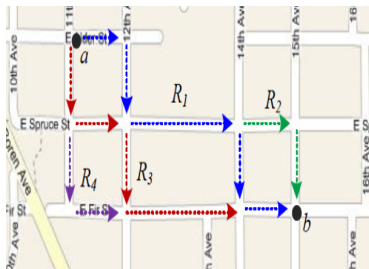


Figure 1: Examples of challenges

Agenda

- 1 Introduction
- 2 Preliminary Processing
- 3 Landmark Graph
- 4 Travel Time Estimation



Preliminary Processing: Data Description

- Is collected from Computational Sensing Lab at Tsinghua University
- Contains 83 million GPS records from 8,602 taxis in Beijing during May of 2009

Field	Explanation
CUID	ID for each taxi
UNIX_EPOCH	Unix timestamp
GPS_LONG	Longitude in WGS-84
GPS_LAT	Latitude in WGS-84
HEAD	Heading direction
SPEED	Instantaneous speed (m/s)
OCCUPIED	Hired (1) or not (0)

Table 1: A summary of the seven original fields

Preliminary Processing: Reverse Geocoding



Preliminary Processing: Reverse Geocoding

- GPS coordinate translation: 1.34°N , $103.68^{\circ}\text{E} \rightarrow$ SCSE, NTU



Preliminary Processing: Reverse Geocoding

- GPS coordinate translation: 1.34°N , 103.68°E \rightarrow SCSE, NTU
- China GPS shift problem: WGS84 v.s. BD09



Figure 2: An example of China GPS shift problem

Preliminary Processing: Reverse Geocoding

- GPS coordinate translation: 1.34°N , 103.68°E \rightarrow SCSE, NTU
- China GPS shift problem: WGS84 v.s. BD09
- Solution: WGS84 $\xrightarrow{\text{Baidu API}}$ BD09 $\xrightarrow{\text{Baidu API}}$ Street



Figure 2: An example of China GPS shift problem

Preliminary Processing: Outlier Detection

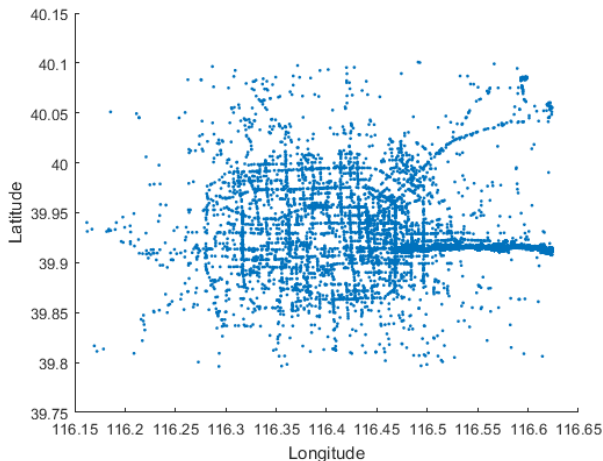


Figure 3: An example of outliers



**NANYANG
TECHNOLOGICAL
UNIVERSITY**
SINGAPORE

Theorem (*Majority Clustering Theorem*)

If a **reasonable reverse geocoder** is used to reverse-geocode a set of GPS data points which are mapped to a particular street *in reality*, then, when plotted on a 2-D plane, majority (more than 50%) of the points must be clustered together to form a rough shape that is similar to the shape of the street that they are supposed to be mapped to.

Theorem (*Majority Clustering Theorem*)

If a **reasonable reverse geocoder** is used to reverse-geocode a set of GPS data points which are mapped to a particular street *in reality*, then, when plotted on a 2-D plane, majority (more than 50%) of the points must be clustered together to form a rough shape that is similar to the shape of the street that they are supposed to be mapped to.

Two-step procedure:

Outlier Detection = Outlier Identification + Outlier Removal

Outlier Identification: Clustering

Outlier Identification: Clustering

- Sample point concentration \rightarrow cluster concentration

Outlier Identification: Clustering

- Sample point concentration \rightarrow cluster concentration
- Top $k\%$ ($k = 50$) largest clusters as groups of correct sample points

Outlier Identification: Clustering

- Sample point concentration \rightarrow cluster concentration
- Top $k\%$ ($k = 50$) largest clusters as groups of correct sample points
- 10×10 self-organising feature maps implementation

Outlier Identification: Clustering

- Sample point concentration \rightarrow cluster concentration
- Top $k\%$ ($k = 50$) largest clusters as groups of correct sample points
- 10×10 self-organising feature maps implementation

Outlier Removal: Distance Threshold d_{max}

Outlier Identification: Clustering

- Sample point concentration \rightarrow cluster concentration
- Top $k\%$ ($k = 50$) largest clusters as groups of correct sample points
- 10×10 self-organising feature maps implementation

Outlier Removal: Distance Threshold d_{max}

- Assign sample points to legal centroids no farther than d_{max}

Outlier Identification: Clustering

- Sample point concentration \rightarrow cluster concentration
- Top $k\%$ ($k = 50$) largest clusters as groups of correct sample points
- 10×10 self-organising feature maps implementation

Outlier Removal: Distance Threshold d_{max}

- Assign sample points to legal centroids no farther than d_{max}
- Remove all “orphan” sample points

Outlier Identification: Clustering

- Sample point concentration \rightarrow cluster concentration
- Top $k\%$ ($k = 50$) largest clusters as groups of correct sample points
- 10×10 self-organising feature maps implementation

Outlier Removal: Distance Threshold d_{max}

- Assign sample points to legal centroids no farther than d_{max}
- Remove all “orphan” sample points
- Use real physical distance on the Earth

Outlier Identification: Clustering

- Sample point concentration \rightarrow cluster concentration
- Top $k\%$ ($k = 50$) largest clusters as groups of correct sample points
- 10×10 self-organising feature maps implementation

Outlier Removal: Distance Threshold d_{max}

- Assign sample points to legal centroids no farther than d_{max}
- Remove all “orphan” sample points
- Use real physical distance on the Earth
- Set $d_{max} = 30\text{m}$ or 50m

Preliminary Processing: Outlier Detection

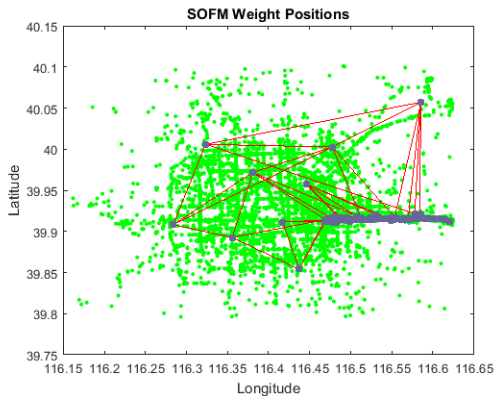


Figure 4: A plot of neuron positions after training



**NANYANG
TECHNOLOGICAL
UNIVERSITY**
SINGAPORE

Preliminary Processing: Outlier Detection

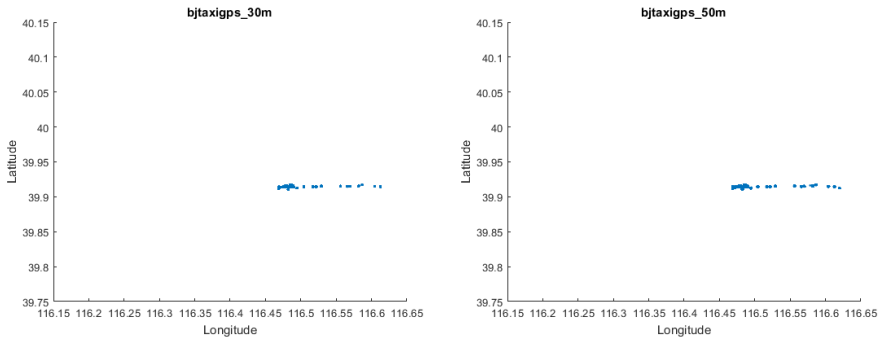


Figure 5: A plot of sample points after outlier removal

Agenda

- 1 Introduction
- 2 Preliminary Processing
- 3 Landmark Graph
- 4 Travel Time Estimation



**NANYANG
TECHNOLOGICAL
UNIVERSITY**
SINGAPORE

Landmark Graph: Basic Ideas

Definition (*Landmark*)

A landmark is a road segment that is frequently traversed by taxi drivers according to the taxi GPS trajectory database.

Definition (*Landmark*)

A landmark is a road segment that is frequently traversed by taxi drivers according to the taxi GPS trajectory database.

Rationale

- Sample points too sparse to model every street accurately

Definition (*Landmark*)

A landmark is a road segment that is frequently traversed by taxi drivers according to the taxi GPS trajectory database.

Rationale

- Sample points too sparse to model every street accurately
- Landmarks resemble common human thinking

Landmark Graph: Basic Ideas

Definition (*Landmark*)

A landmark is a road segment that is frequently traversed by taxi drivers according to the taxi GPS trajectory database.

Rationale

- Sample points too sparse to model every street accurately
- Landmarks resemble common human thinking

Step to build landmark graph

- Separate sample points into **trips**

Landmark Graph: Basic Ideas

Definition (*Landmark*)

A landmark is a road segment that is frequently traversed by taxi drivers according to the taxi GPS trajectory database.

Rationale

- Sample points too sparse to model every street accurately
- Landmarks resemble common human thinking

Step to build landmark graph

- Separate sample points into **trips**
- Count occurrences of each street

Landmark Graph: Basic Ideas

Definition (*Landmark*)

A landmark is a road segment that is frequently traversed by taxi drivers according to the taxi GPS trajectory database.

Rationale

- Sample points too sparse to model every street accurately
- Landmarks resemble common human thinking

Step to build landmark graph

- Separate sample points into **trips**
- Count occurrences of each street
- Find connections between two landmarks



Landmark Graph: Trip Identification

CUID	UTC	GPS_LONG	GPS_LAT	OCCUPIED	TRIP_ID
1	1/5/2009 0:02:00	116.39616	39.81294	0	4552265
1	1/5/2009 0:04:00	116.39575	39.82296	0	4552265
1	1/5/2009 0:07:00	116.39567	39.82774	0	4552265
1	1/5/2009 17:08:00	116.30142	39.98105	1	1
1	1/5/2009 17:10:00	116.29514	39.98419	1	1
1	1/5/2009 17:11:00	116.28959	39.98289	1	1
1	1/5/2009 17:12:00	116.28087	39.97552	1	1
1	1/5/2009 17:16:00	116.26813	39.93537	1	1
1	1/5/2009 18:11:00	116.36537	39.95019	0	4552271
1	1/5/2009 18:12:00	116.36546	39.94886	0	4552271
1	1/5/2009 18:13:00	116.35927	39.94528	0	4552271

Table 2: An example of trip identification



**NANYANG
TECHNOLOGICAL
UNIVERSITY**
SINGAPORE

Landmark Graph: Frequency Counting

CUID	UTC	GPS_LONG	GPS_LAT	Street	TRIP_ID
1	1/5/2009 0:02:00	116.39616	39.81294	A	4552265
1	1/5/2009 0:04:00	116.39575	39.82296	A	4552265
1	1/5/2009 0:07:00	116.39567	39.82774	B	4552265
1	1/5/2009 17:08:00	116.30142	39.98105	C	1
1	1/5/2009 17:10:00	116.29514	39.98419	C	1
1	1/5/2009 17:11:00	116.28959	39.98289	C	1
1	1/5/2009 17:12:00	116.28087	39.97552	A	1
1	1/5/2009 17:16:00	116.26813	39.93537	A	1
1	1/5/2009 18:11:00	116.36537	39.95019	B	4552271
1	1/5/2009 18:12:00	116.36546	39.94886	C	4552271
1	1/5/2009 18:13:00	116.35927	39.94528	C	4552271

Table 3: An illustration of frequency counting



**NANYANG
TECHNOLOGICAL
UNIVERSITY**
SINGAPORE

Landmark Graph: Construction

For each trip

- Select a landmark j
- Record intermediate streets while searching for the next landmark k
- Repeat the process starting from k until all streets are examined

Agenda

- 1 Introduction
- 2 Preliminary Processing
- 3 Landmark Graph
- 4 Travel Time Estimation



Definition (*Significant Edge*)

A significant edge in a landmark graph $G = (V, E)$ is an edge $e \in E$ that has a support at least m , where m is a parameter specified in advance.

Definition (*Significant Edge*)

A significant edge in a landmark graph $G = (V, E)$ is an edge $e \in E$ that has a support at least m , where m is a parameter specified in advance.

- Build a **predictive model** for travel time of each significant edge

Definition (*Significant Edge*)

A significant edge in a landmark graph $G = (V, E)$ is an edge $e \in E$ that has a support at least m , where m is a parameter specified in advance.

- Build a **predictive model** for travel time of each significant edge
- Separate weekday's travel time from weekend's

Definition (*Significant Edge*)

A significant edge in a landmark graph $G = (V, E)$ is an edge $e \in E$ that has a support at least m , where m is a parameter specified in advance.

- Build a **predictive model** for travel time of each significant edge
- Separate weekday's travel time from weekend's
- Evaluate results against Baidu's estimates

Travel Time Estimation: Underlying Distribution

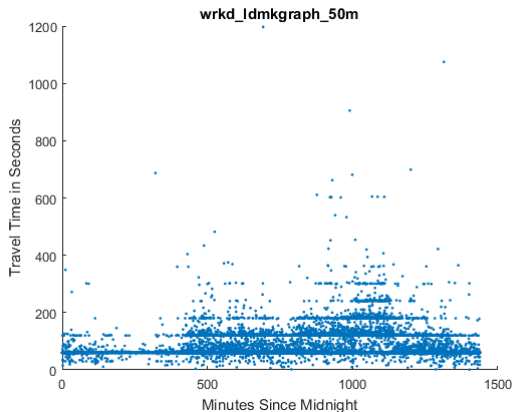


Figure 6: An example of travel time patterns

Travel Time Estimation: Underlying Distribution

Possible Explanations

- Drivers choose different routes to travel between the two landmarks

Travel Time Estimation: Underlying Distribution

Possible Explanations

- Drivers choose different routes to travel between the two landmarks
- Drivers have different driving skills, preferences and behaviours



Travel Time Estimation: Underlying Distribution

Possible Explanations

- Drivers choose different routes to travel between the two landmarks
- Drivers have different driving skills, preferences and behaviours
- The GPS devices report locations **periodically**, therefore, durations like 60 seconds or 120 seconds are very common

Travel Time Estimation: Clustering

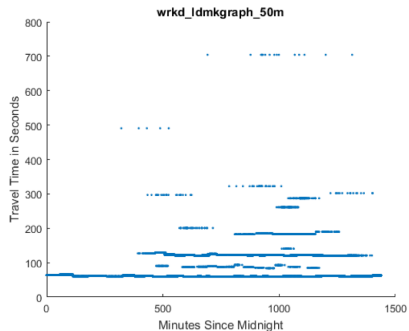
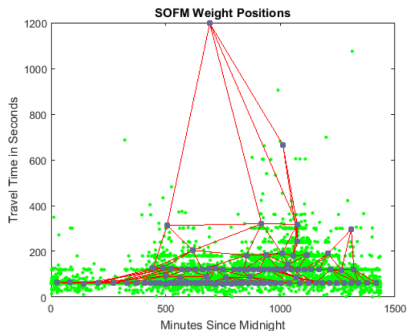


Figure 7: An illustration of travel time clustering