When Will You Arrive?
Estimating Travel Time Based on Deep Neural Networks

Dong Wang[1], Junbo Zhang[2], Wei Cao[3], Jian Li[3], Yu Zheng[245]

[1]Duke University, [2]Microsoft Research, [3]Tsinghua University, [4]Xidian University [5]Chinese Academy of Sciences

Introduction

Objective:

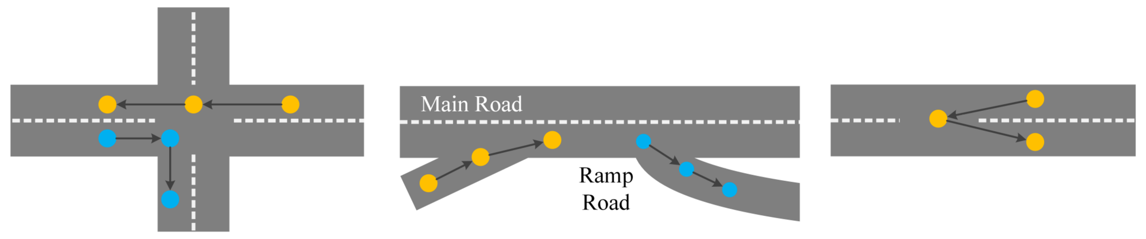
Given: path (sequence of locations), start time, driver(optional)
Estimate: the travel time

Background:

Estimating the travel time in a city is of great importance to *traffic monitoring*, *route planning*, *ridesharing*, *taxi/Uber dispatching*, etc.

Challenges:

1. Diverse complex factors:
   - Spatial and temporal dependencies
   - External factors (weather, driver habit, day of the week)
2. Balance between collective vs individual
   - Estimate the travel time of each individual roads (Error accumulates)
   - Estimate the travel time of the entire path (Data sparsity problem)
3. Data sparsity problem
   - Over 80% GPS records fall on <20% roads



(a) Go straight (yellow) or turn right (blue) (b) Drive into main road (yellow) or ramp road (blue) (c) Turn around

Data:

GPS trajectory: a sequence of GPS points
GPS point: latitude, longitude, timestamp, driver ID(optional).
Sample a GPS point each 200m ~ 400m
We use the timestamp as the **ground truth**.

Multi-task Learning Component

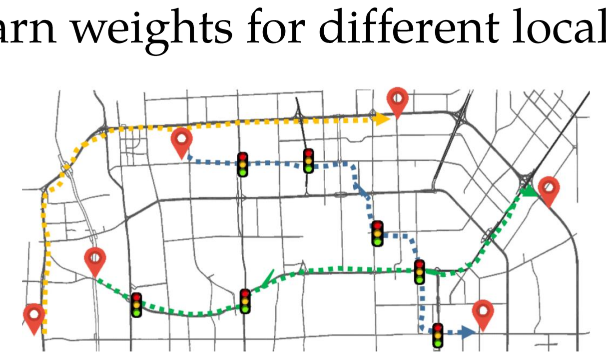
✓ **Estimate the local path:**

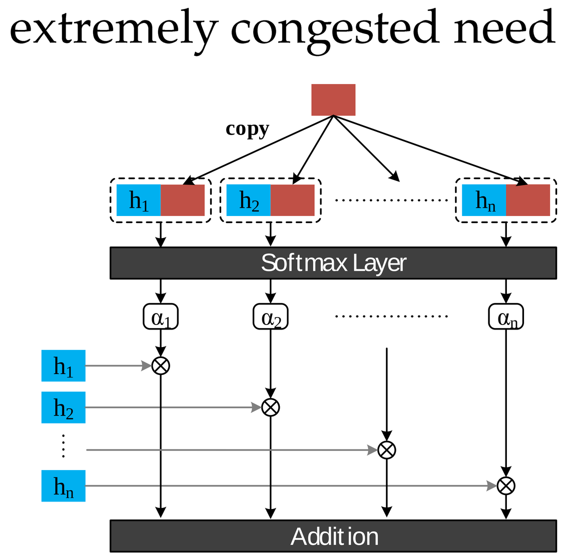
Spatio-temporal feature sequence of local paths: $\{h_1, h_2, \ldots, h_{|T|-k+1}\}$.

✓ **Estimate the entire path:** $h_{att} = \sum_{i=1}^{|T|-k+1} \alpha_i \cdot h_i$

Attention Mechanism:

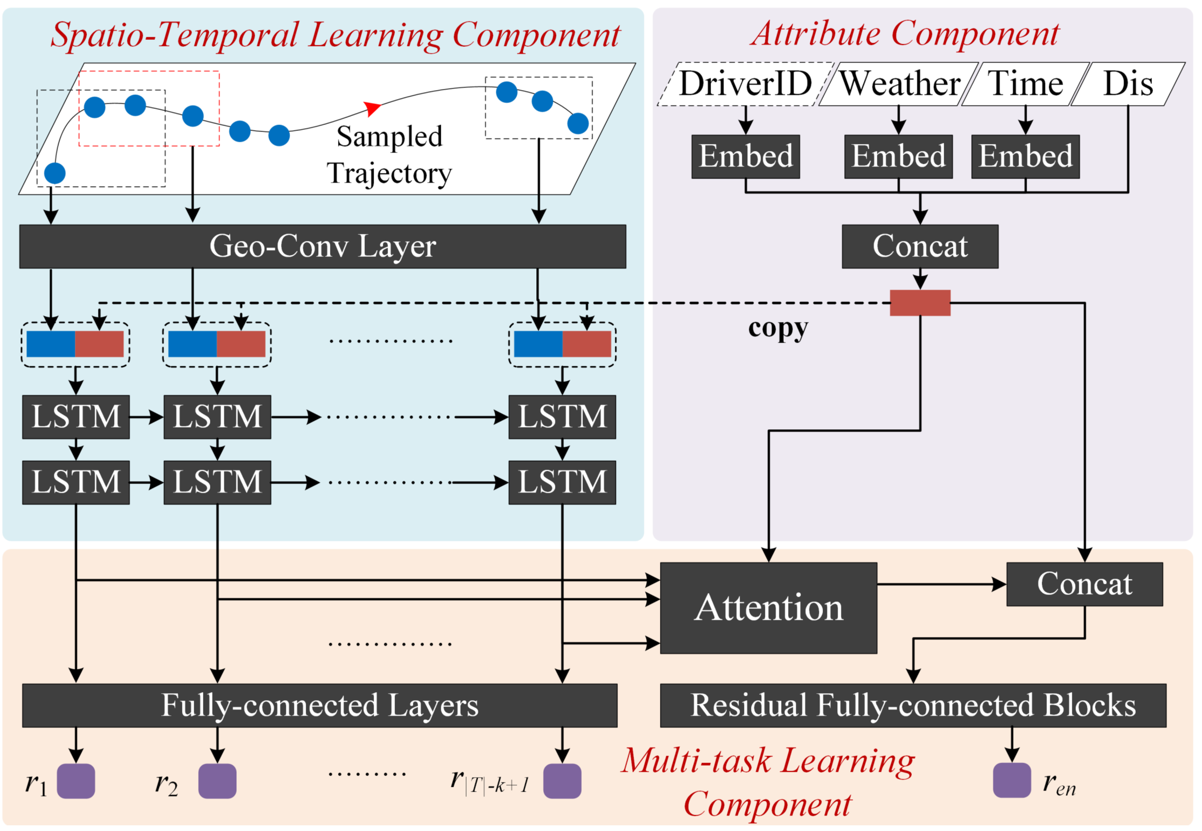
- Local path with more intersections or in extremely congested need more attention.
- Learn weights for different local path





Training Loss: $\beta \cdot L_{local} + (1-\beta) \cdot L_{en}$

DeepTTE Architecture

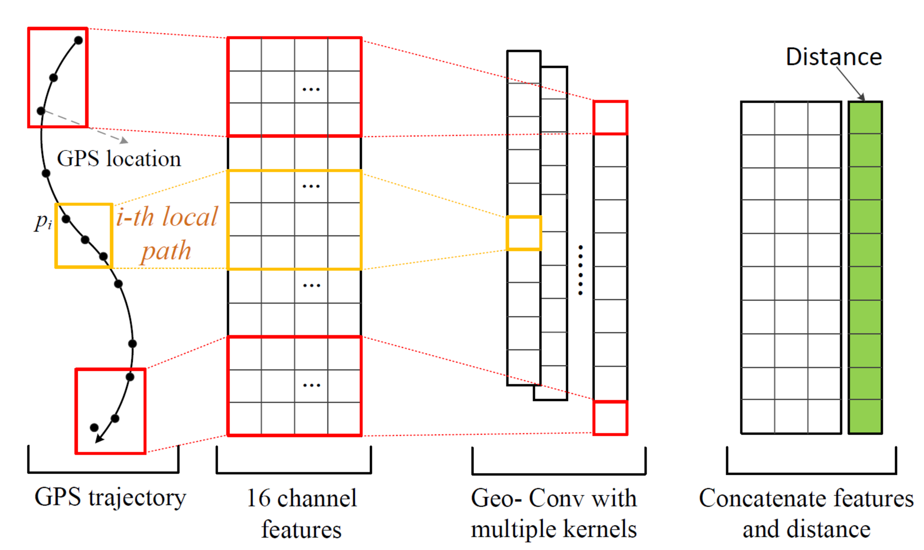


1. Spatio-temporal Learning Component:
   - Geo-Conv Layer: Capture spatial correlations
   - Recurrent Layer: Learn temporal dependencies
2. Attribute Component:
   - Handle external factors & share similar pattern
3. Multi-task Learning Component:
   - Address data sparsity problem
   - Get a better estimation result

Spatio-temporal Learning Component

Geo-Conv Layer

Transforms the raw GPS sequence to a series of feature maps for each local paths.



$$loc_i = \tanh(W_{loc} \cdot [p_i.lat \circ p_i.lng]) \quad (1)$$

$$loc_i^{conv} = \sigma_{cnn}(W_{conv} * loc_{i:i+k-1} + b) \quad (2)$$

Recurrent Layer

To further capture the temporal dependencies among these local paths, we introduce the recurrent layers in our model.

$$h_i = \sigma_{rnn}(W_x \cdot loc_i^f + W_h \cdot h_{i-1} + W_a \cdot attr) \quad (3)$$

Summary

1. We propose an end-to-end Deep learning framework for Travel Time Estimation (called DeepTTE).
2. We present a **geo-convolution operation** by integrating the geographic information into the classical convolution, capable of capturing spatial correlations for trajectory data.
3. We propose a multi-task component to balance the individual and collective estimations, and address data sparsity problem.
4. Our model considers various factors which may affect the travel time: the weather, driver habit, and road information, etc.
5. We conduct extensive experiments on **two very large scale real-world datasets**. The results show that our model outperforms the other off-the-shell methods significantly.

Code and sample data: https://github.com/UrbComp/DeepTTE

Experiment

Dataset:

Chengdu dataset: consists of *9,737,557 trajectories* (1.4 billion GPS records) of 14,864 taxis in August 2014 in Chengdu, China.
Beijing dataset: consists of 3,149,023 trajectories (0.45 billion GPS records) of 20,442 taxis in April 2015 in Beijing, China.
** For Beijing Dataset, we further collected the corresponding weather conditions (16 types including sunny, rainy, cloudy etc.) as well as the road ID of each GPS point.*
* We use the last 7 days in each datasets as the test set.

	Chengdu		Beijing	
	MAPE (%)	RMSE (sec)	MAPE (%)	RMSE (sec)
AVG	28.1	533.57	24.78	703.17
D-TEMP	22.82	441.50	19.63	606.76
GBDT	19.32 ± 0.04	357.09 ± 2.44	19.98 ± 0.02	512.96 ± 3.96
MlpTTE	16.90 ± 0.06	379.39 ± 1.94	23.73 ± 0.14	701.61 ± 1.82
RnnTTE	15.65 ± 0.06	358.74 ± 2.02	13.73 ± 0.05	408.33 ± 1.83
DeepTTE	$\mathbf{11.89 \pm 0.04}$	$\mathbf{282.55 \pm 1.32}$	$\mathbf{10.92 \pm 0.06}$	$\mathbf{329.65 \pm 2.17}$

Tab: Performance Comparison

TEMP [1] is the state-of-art collective estimation method. However, there are about 10% paths that the original TEMP method can not estimate due to the lack of neighbor trajectories. We refine it as D-TEMP.

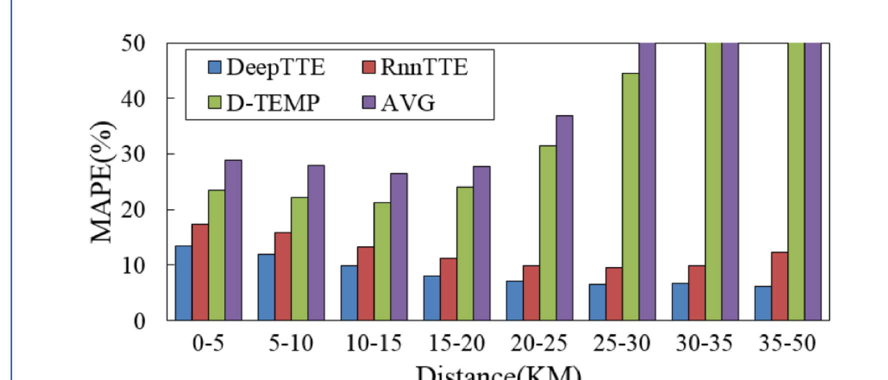


Fig: Error rates for trajs with different lengths

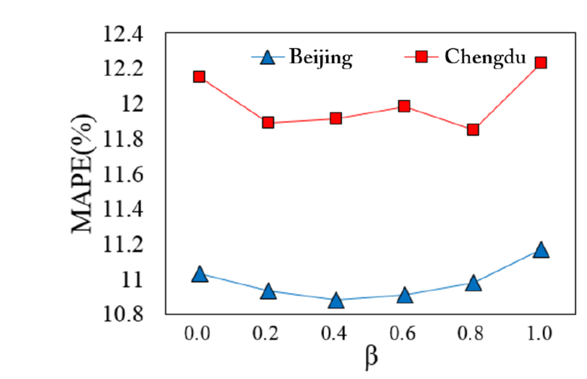


Fig: Error rates for different β

Effect of Attribute Component:
D-TEMP: 19.63%, 5-MLP : 23.73%, 1-MLP+Attri: 17.63%
Effect of Geo-Conv Layer:
Without Geo-Conv layer: 13:14% and 12:68%; With 11:89% and 10:92%).

Contact

Dong Wang
Duke University
Email: dong.wang363@duke.edu
Website: https://jelly007.github.io/

Wei Cao
Tsinghua University
Email: cao-w13@mails.Tsinghua.edu.cn
Website: https://caow13.github.io/

Code

References

1. Wang, H.; Kuo, Y.-H.; Kifer, D.; and Li, Z. A simple baseline for travel time estimation using large-scale trip data. In Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 61. ACM.
2. Wang, Y.; Zheng, Y.; and Xue, Y. 2014. Travel time estimation of a path using sparse trajectories. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, 25–34. ACM.
3. Yang, B.; Guo, C.; and Jensen, C. S. 2013. Travel cost inference from sparse, spatio temporally correlated time series using markov models. Proceedings of the VLDB Endowment 6(9):769–780.
4. Yuan, J.; Zheng, Y.; Xie, X.; and Sun, G. 2013. T-drive: enhancing driving directions with taxi drivers' intelligence. Knowledge and Data Engineering, IEEE Transactions on 25(1):220–232.
5. Zhang, J.; Zheng, Y.; and Qi, D. 2017. Deep spatio-temporal residual networks for citywide crowd flows prediction. In AAAI, 1655–1661.
6. Wang, D.; Cao, W.; Ye, P.; and Li, J. DeepSD: supply-demand prediction for online car-hailing services using deep neural networks. In Data Engineering (ICDE), 2017 IEEE 33rd International Conference on.