Instruction-Tuning Llama-3-8B Excels in City-Scale Mobility Prediction

Peizhi Tang[1], **Chuang Yang[2]**, Tong Xing[2],
Xiaohang Xu[2], Renhe Jiang[2], Kaoru Sezaki[2]

Southern University of Science and Technology[1], The University of Tokyo[2]

HuMob'24@SIGSPATIAL
Oct 29th, 2024

I. Introduction

➢ Motivation

Typical Procedures

New Paradigm

Data Preprocessing

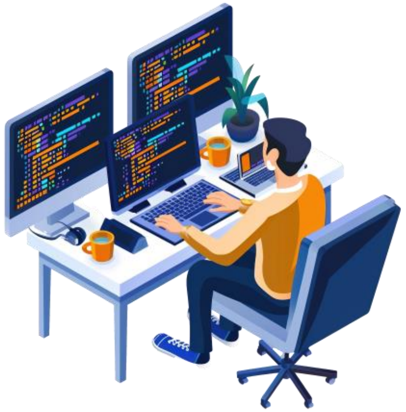
missing inconsistency

ID	Name	Date of Birth	Prefecture	Postal Code	Height
1	Yuka	2003/02/26	Hokkaido	540-8570	165
2	Nana		Aichi	464-0804	157
3	Nana	2003/03/30	Aichi	464-0804	157
4	Miho	2001/06/25	Kangawa	2208799	1.60

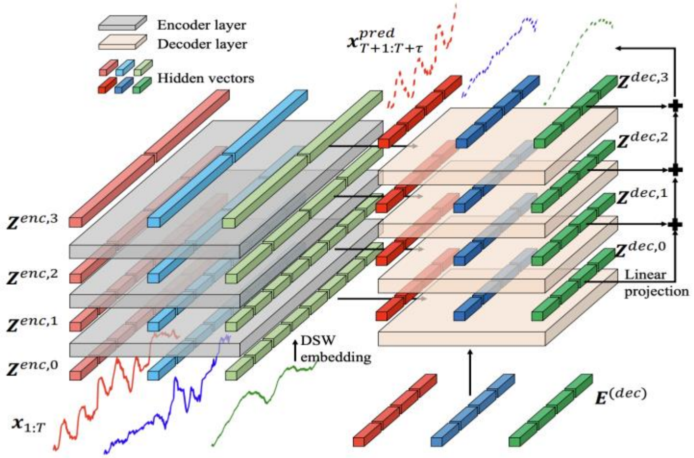
duplicate typo format outlier

ID	Name	Date of Birth	Prefecture	Postal Code	Height
1	Yuka	2003/02/26	Osaka	540-8570	165
2	Nana	2003/03/30	Aichi	464-0804	157
4	Miho	2001/06/25	Kanagawa	220-8799	160

Zhang et al., EMNLP 2024



Time Series Prediction

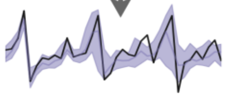


Zhang et al., ICLR 2023

"631, 656, 650, ..., 487, 485, 487"

LLM

"479, ..., 371, 364"
"492, ..., 499, 501"

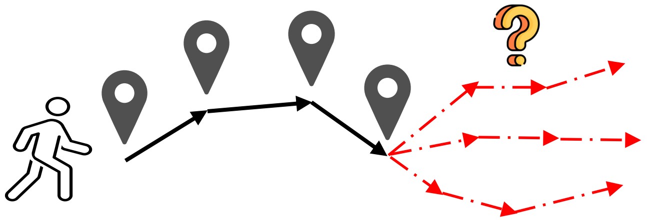


Gruver et al., NeurIPS 2024

[1] Zhang et al., "Jellyfish: A Large Language Model for Data Preprocessing" *EMNLP 2024*.
[2] Zhang et al., " Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting", ICLR 2023.
[3] Gruver et al., "Large Language Models Are Zero-Shot Time Series Forecasters", NeurIPS 2023.

I. Introduction

➢ Motivation



Human Mobility Prediction

Typical Procedures

Reference	Name	Year	DL Modules	Evaluation
Abideen et al. [1]	DWSTTN	2021	Encoder, Decoder, Attention, FC	Distance
Tang et al. [186]	CLNN	2021	LSTM, Embedding, FC	Distance
Bao et al. [10]	BiLSTM-CNN	2020	Embedding, BiLSTM, CNN	ACC@k
Chen et al. [36]	DeepJMT	2020	GRU, FC, Encoder	ACC@k
Yang et al. [217]	Flashback	2020	Attention, RNN	ACC@k
Ebel et al. [52]	-	2020	RNN, FC, Embedding	Distance
Rossi et al. [156]	-	2019	Attention, LSTM	Distance
Gao et al. [67]	VANext	2019	CNN, GRU, Attention	ACC@k
Kong et al. [103]	HST-LSTM	2018	LSTM	ACC
Lv et al. [122]	T-CONV	2018	CNN, FC	Distance
Feng et al. [57]	DeepMove	2018	Attention, GRU, FC	ACC
Yao et al. [220]	SERM	2017	LSTM	ACC@k
Liu et al. [118]	ST-RNN	2016	RNN	Rec@k, F1@k, MAPE, AUC
De Brébisson et al. [47]	-	2015	FC	Distance

Luca et al., CSUR 2021

New Paradigm

Can we predict human mobility through LLMs in a Q&A manner ?

Question

Here is the historical trajectories of a user, the format of each record is <day_id, timeslot_id, x, y>:
<0,0,199,199>,<0,1,198,196>, <0,2,195,196>, ⋯<59,47,198,196>, <60,1,999,999>,<60,2,999,999>,<61,1,999,999>,⋯<74,47,999,999>.

- <day_id, timeslot_id> depicts the time information,
- <x, y> show the coordinates

please predict the location masked with 999 and give me the reason.

Answer

OK, No Problem.

- The predicted trajectories is: [[60,1,92,94], [60,2,93,95], [61,1,93,94],⋯ [74,47,101,108]]
- The reason is that the individual's trajectory shows a consistent pattern, likely to follow the established pattern...

[1] Luca et al., "A survey on deep learning for human mobility" CSUR *2021*.

II. Selection of Technical Route

➢ Comparation of Open-source LLMs and Closed-source LLMs

Closed-source LLMs

Open-source LLMs

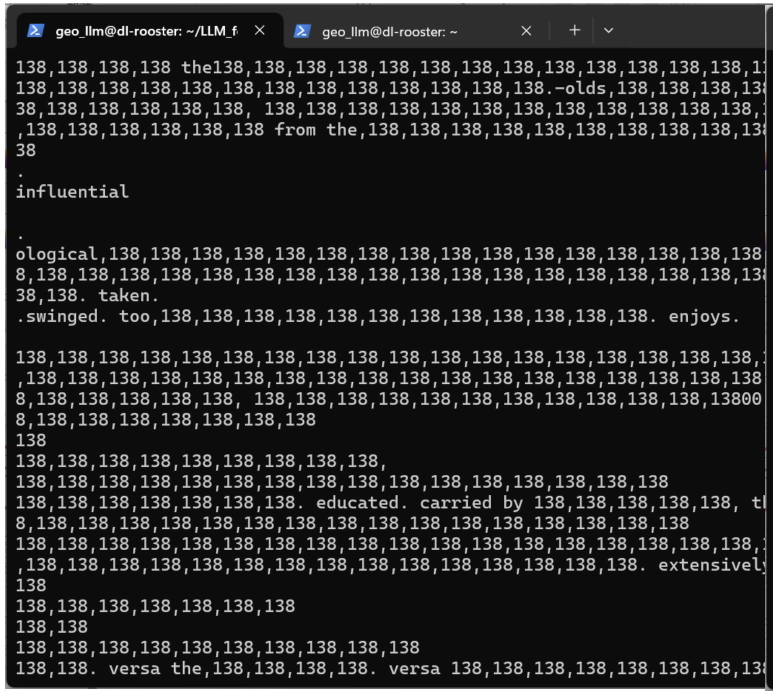
Data Leakage

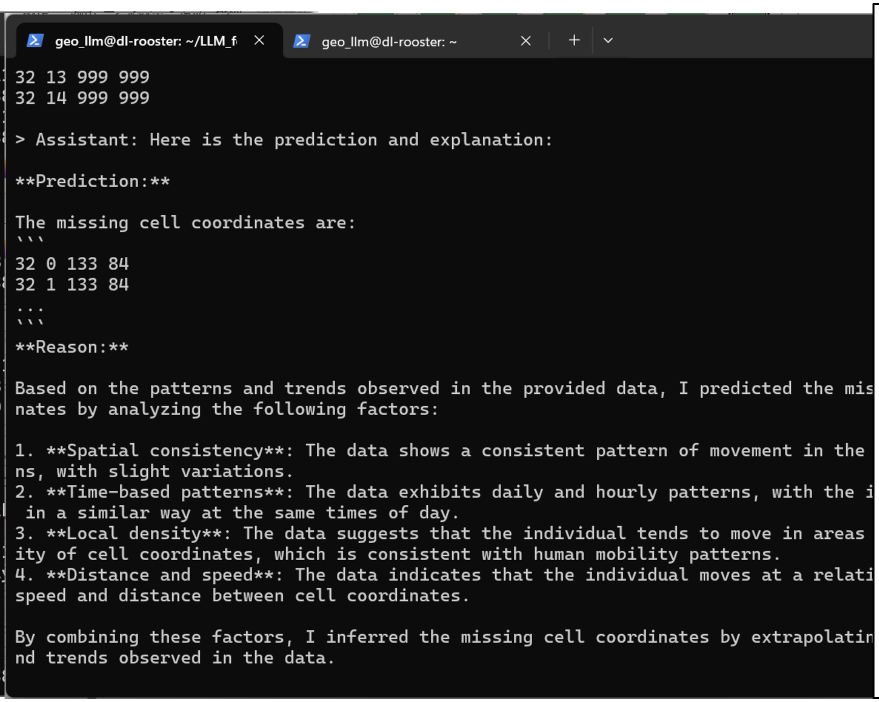
Cost

Reproduceable

II. Selection of Technical Route

➢ Drawbacks of LLaMA.



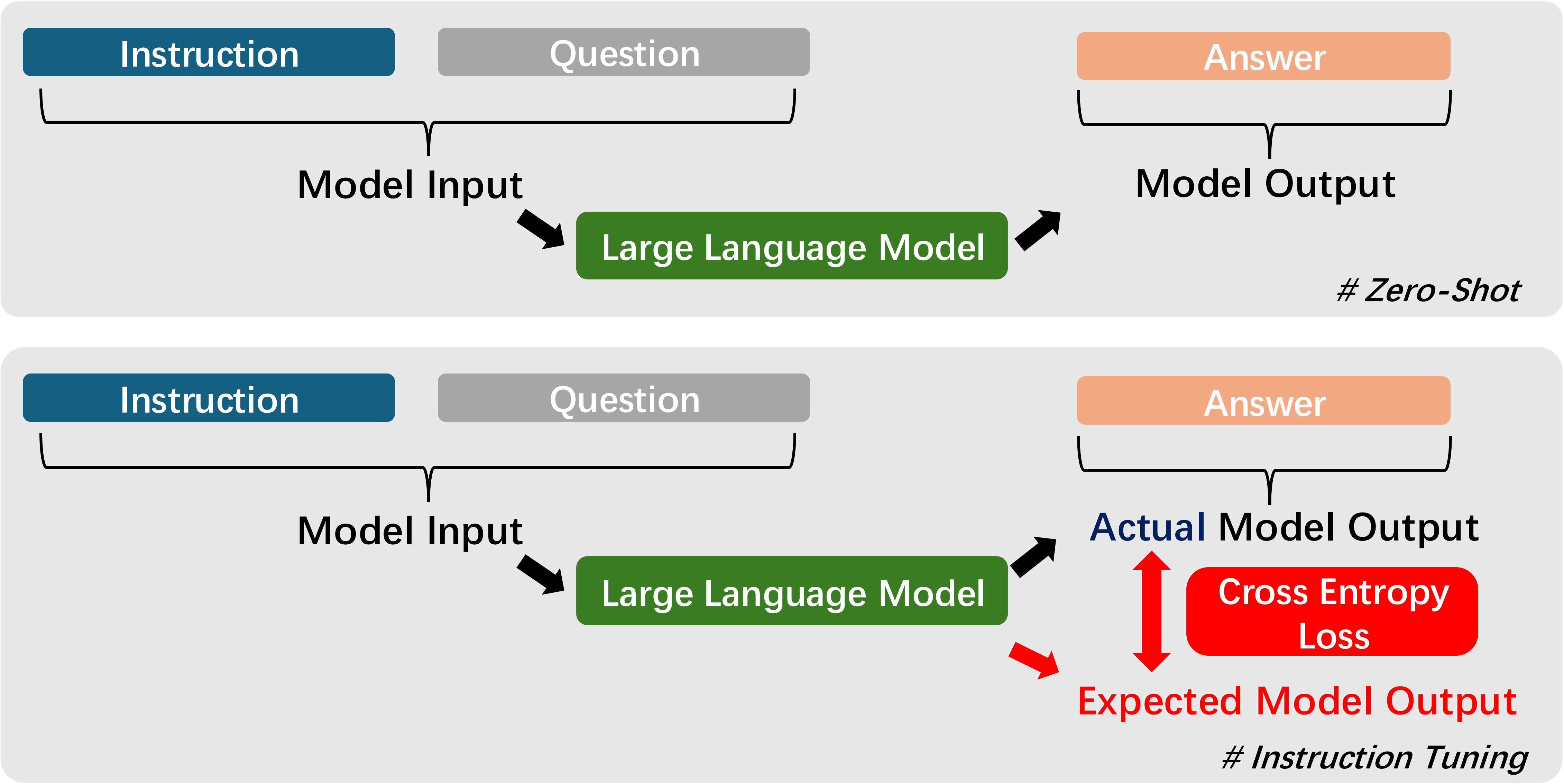


Hallucination | Incomplete output | Incorrect format

Poor Zero-shot ability for human mobility prediction

III. Proposed Method

➢ From Zero-Shot to Instruction Tuning.



III. Proposed Method

➢ Instruction Designing

Instruction

[Role] You are a helpful assistant that predicts human mobility trajectories in a city.

[Introduction *#Environment*]

- The target city is divided into equally sized cells, creating a 200 x 200 grid.
- We use coordinate **<x>,<y>** to indicate the location of a cell within the target area.
- The horizontal coordinate **<x>** increases from left to right, and the vertical coordinate **<y>** increases from top to bottom.
- The coordinates of the top-left corner are (0, 0), and the coordinates of the bottom-right corner are (199, 199).

[Introduction *#Trajectory Definition*]

- A trajectory is a sequence of quadruples ordered by time.
- Each quadruple follows the format **<day_id>, <timeslot_id>, <x>, <y>**.
  It represents a person's location **<x>, <y>** at the timeslot **<timeslot_id>** of day **<day_id>**.
- The **<day_id>** is the index of day, representing a specific day.
- Each day's 24 hours are discretized into 48 time slots with a time interval of 30 minutes.
  <timeslot_id> is the index of the time slot, ranging from 0 to 47, representing a specific half-hour in a day.

[Introduction *#Trajectory Example*]

- Let me give you an example of a quadruple to better illustrate what is a record in a trajectory.
- For instance, a sequence (1,12,124,121) indicates that an individual was located at cell 124,121 between 11:30 and 12:00 on day 1.

[#Task Description]

You will receive an individual's trajectory in the target city, with some cell coordinates **<x>,<y>** that were missed and marked as 999,999.
Please **replace all instances of 999 with your predictions**.

[#Format emphasizing]

- Please organize your answer in Json object containing following keys:
  {"prediction": here should be the missing part of sequence only, without adding any extra things.}
- Do not write any code, just inference by yourself.
- Do not provide any other things in your response besides the Json object.

III. Proposed Method

➢ Instruction Designing

Instruction

[Role] You are a helpful assistant that predicts human mobility trajectories in a city.

[Introduction *#Environment*]

- The target city is divided into equally sized cells, creating a 200 x 200 grid.
- We use coordinate **<x>,<y>** to indicate the location of a cell within the target area.
- The horizontal coordinate **<x>** increases from left to right, and the vertical coordinate **<y>** increases from top to bottom.
- The coordinates of the top-left corner are (0, 0), and the coordinates of the bottom-right corner are (199, 199).

[Introduction *#Trajectory Definition*]

- A trajectory is a sequence of quadruples ordered by time.
- Each quadruple follows the format **<day_id>, <timeslot_id>, <x>, <y>**.
  It represents a person's location **<x>, <y>** at the timeslot **<timeslot_id>** of day **<day_id>**.
- The **<day_id>** is the index of day, representing a specific day.
- Each day's 24 hours are discretized into 48 time slots with a time interval of 30 minutes.
  <timeslot_id> is the index of the time slot, ranging from 0 to 47, representing a specific half-hour in a day.

[Introduction *#Trajectory Example*]

- Let me give you an example of a quadruple to better illustrate what is a record in a trajectory.
- For instance, a sequence (1,12,124,121) indicates that an individual was located at cell 124,121 between 11:30 and 12:00 on day 1.

[#Task Description]

You will receive an individual's trajectory in the target city, with some cell coordinates **<x>,<y>** that were missed and marked as 999,999. Please **replace all instances of 999 with your predictions**.

[#Format emphasizing]

- Please organize your answer in Json object containing following keys:
  {"prediction": here should be the missing part of sequence only, without adding any extra things.}
- Do not write any code, just inference by yourself.
- Do not provide any other things in your response besides the Json object.

Question

Here is the data I wish you to predict:
day_id, timeslot_id, x, y
<0,0,199,199>,<0,1,198,196>, <0,2,195,196>, …<59,47,198,196>,
<60,1,**999,999**>,<60,2,**999,999**>,<61,1,**999,999**>,…<74,47,**999,999**>.

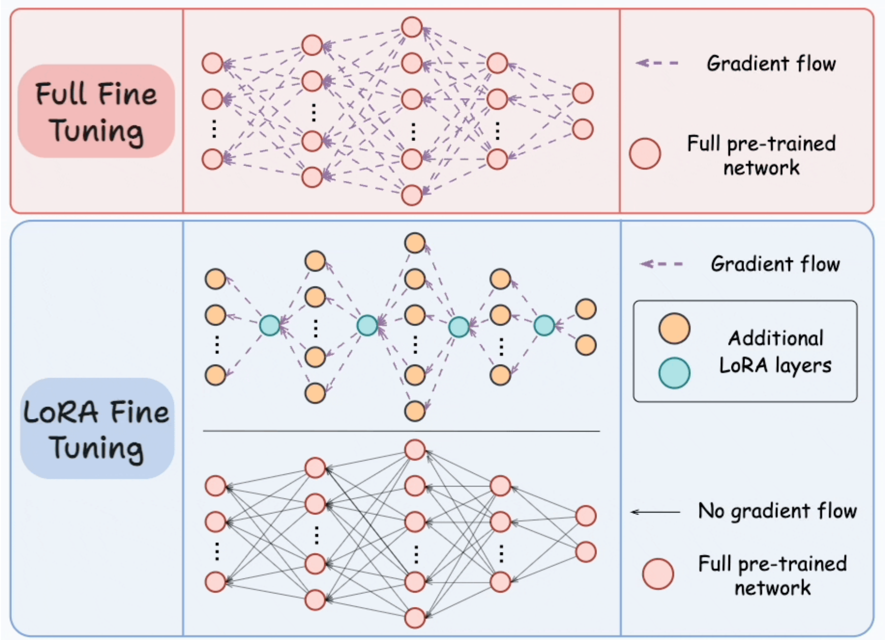
Answer

{ "prediction":
[[60,1,**92,94**], [60,2,**93,95**], [61,1,**93,94**],... [74,47,**101,108**]]
}

III. Proposed Method

➢ Efficient Finetuning

Llama-3-8B



Source: https://www.reddit.com/r/deeplearning/comments/1b6g8ft/full_finetuning_vs_lora_finetuning_vs_rag/#lightbox

Parameters

8 Billons

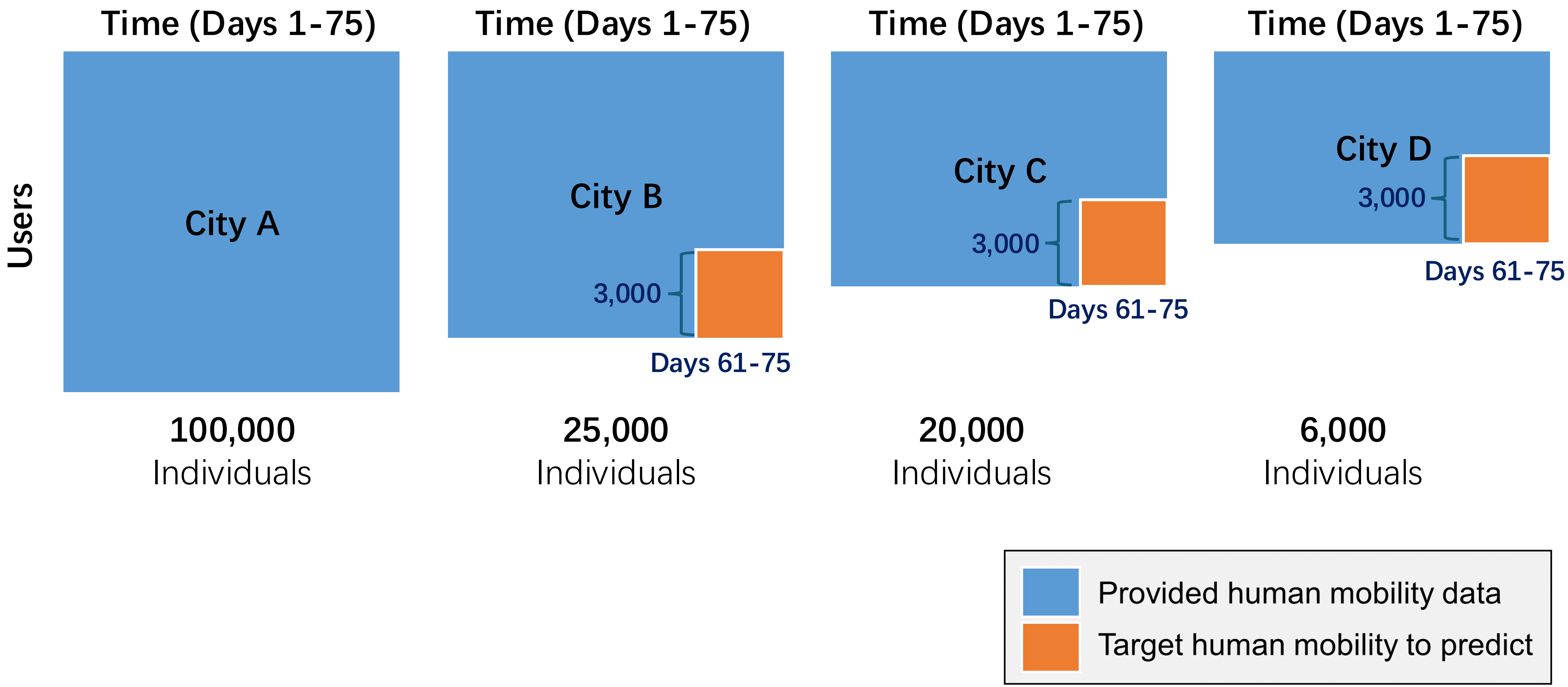
99.95% Reduction

41.94 Millions

Summary of Our Solution: Instruction-Tuning Llama-3-8B with LoRA

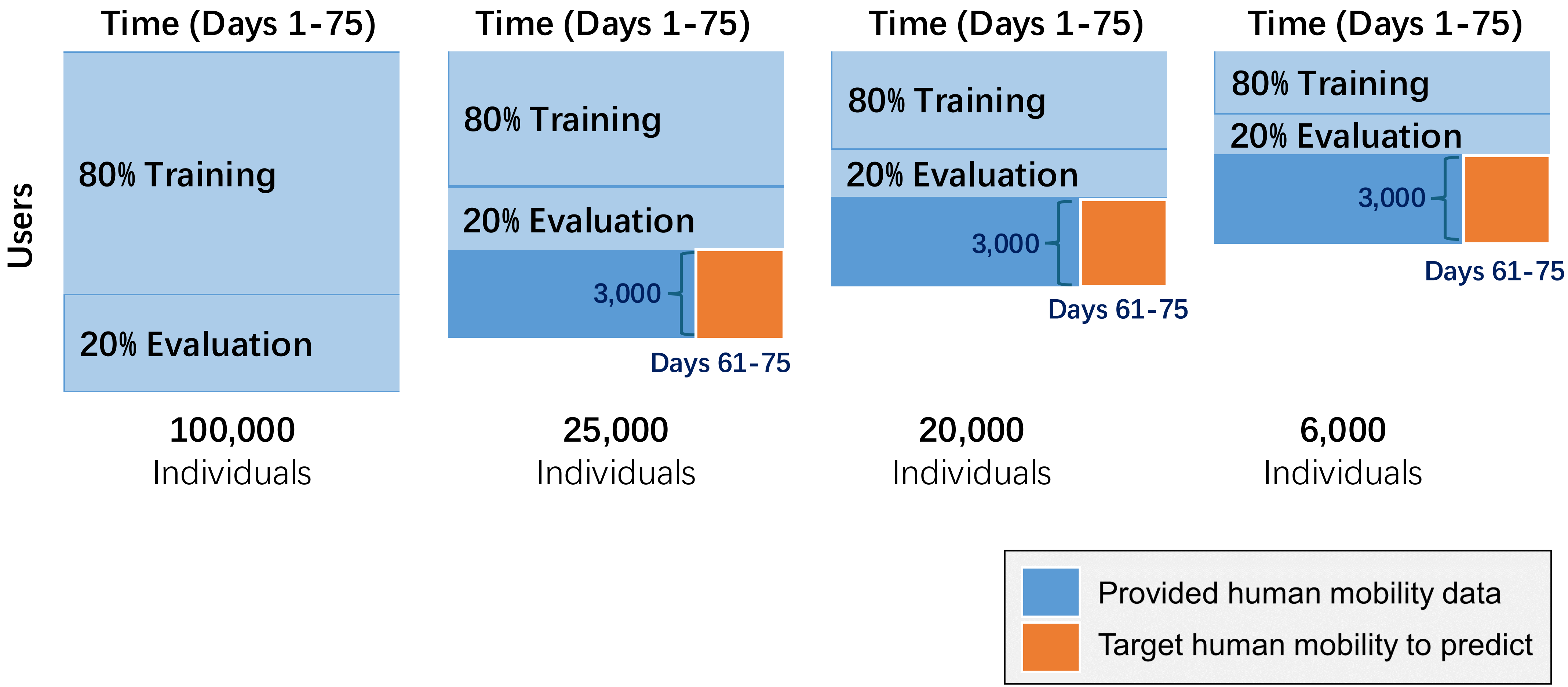
IV. Experiments

➢ Data Preparation



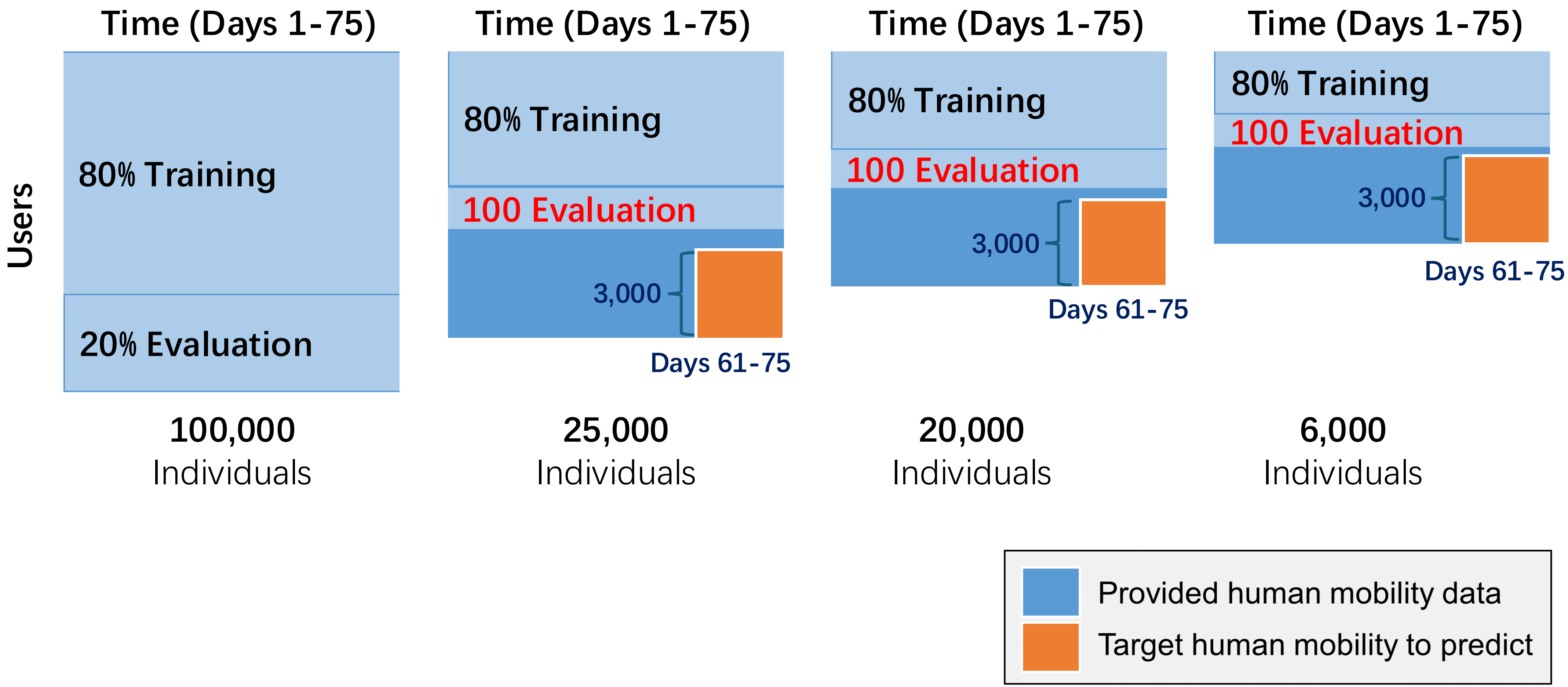
IV. Experiments

➢ Data Preparation



IV. Experiments

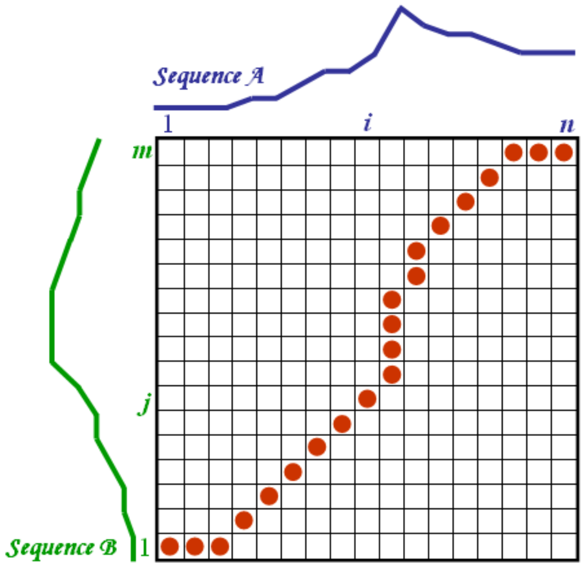
➢ Data Preparation



IV. Experiments

➢ Evaluation Metrics

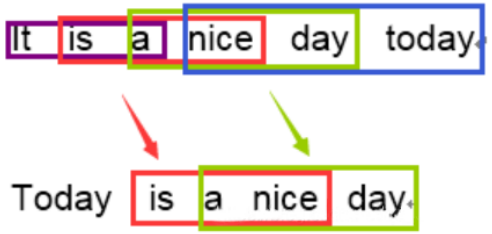
DTW



The shortest cumulative distance

Global Shape Similarity

GEO-BLEU

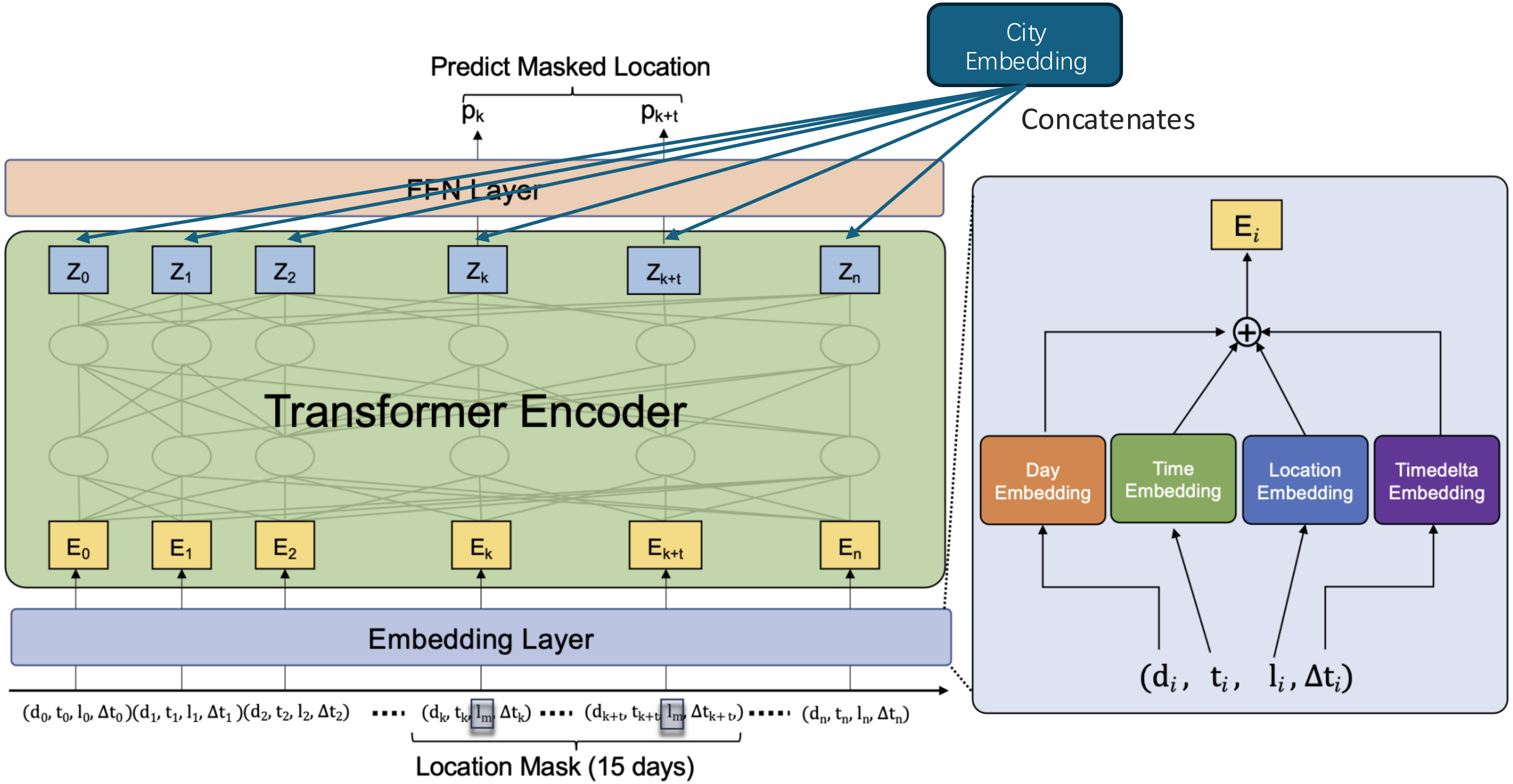


$$GEO\text{-}BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log q_n\right).$$

A geospatial variant of BLEU.
Local Similarity

IV. Experiments

➢ Baseline



Framework of LP-Bert (The Champion of HuMob'23)

IV. Experiments

➢ Results -> Effectiveness Evaluation

- LP-Bert (SOTA): Trained based on training data from all 4 cities.
- Llama-3-8B-Mob *w/ B or C or D*: Fine-tuned with training data of cities B, C, and D, respectively.
- Llama-3-8B-Mob *w/ A+B*: Trained using all training data from B plus additional 1000 users from city A > Covering longer trajectory scenarios.

Model	Average DTW (↓)			Average GEO-BLEU (↑)			Mean Rank	# Trajs Used for Training
	B	*C*	*D*	*B*	*C*	*D*		
LP-Bert [13]	**23.30**	23.81	38.89	0.3093	0.2682	0.3033	4.17	113,600
Llama-3-8B-Mob *w/ B*	26.32	22.49	<u>34.41</u>	0.3322	<u>0.2895</u>	<u>0.3157</u>	2.50	17,600
Llama-3-8B-Mob *w/ C*	31.58	23.75	34.49	<u>0.3399</u>	0.2891	0.2833	3.67	13,600
Llama-3-8B-Mob *w/ D*	28.75	<u>22.20</u>	38.46	0.3251	0.2765	0.3056	3.50	2,400
Llama-3-8B-Mob *w/ A+B*	<u>25.39</u>	**20.57**	**31.94**	**0.3541**	**0.2969**	**0.3217**	**1.17**	18,600

* All experiments were conducted on 4 NVIDIA RTX A6000 48GB GPUs.

- Even a very small amount of data (2,400 users of city D) can surpass the SOTA.
  - Strong zero-shot generalizability across cities.
  - LLMs can mimic human mobility very well with limited data.
- Finetuning with A (1,000) plus B (17,600) achieves the best performance.
- Note: Due to computational efficiency and time constraints of the competition, we merely conducted very limited exploration.

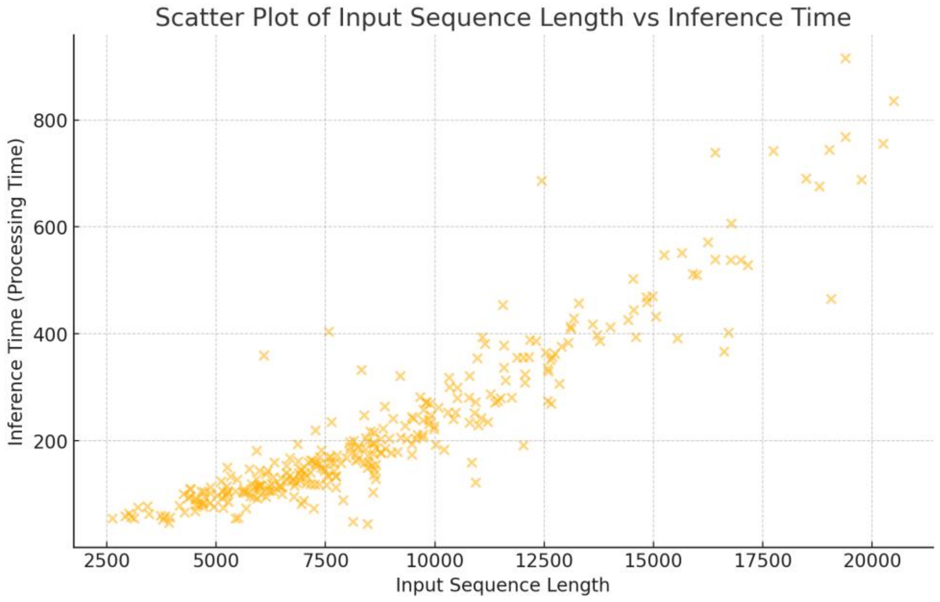
IV. Experiments

➢ Results -> Efficiency Evaluation

Model	# Trainable Parameters	Training		Inference	
		GPU Usage	t_{total}	GPU Usage	t_{infer}
LP-Bert [13]	12.20 M	25.97 GiB	2.77 d	1.49 GiB	13.94 ms
Llama3-8B-Mob *w/ A+B*	41.94 M	43.11 GiB	6.64 d	14.86 GiB	225.61 s

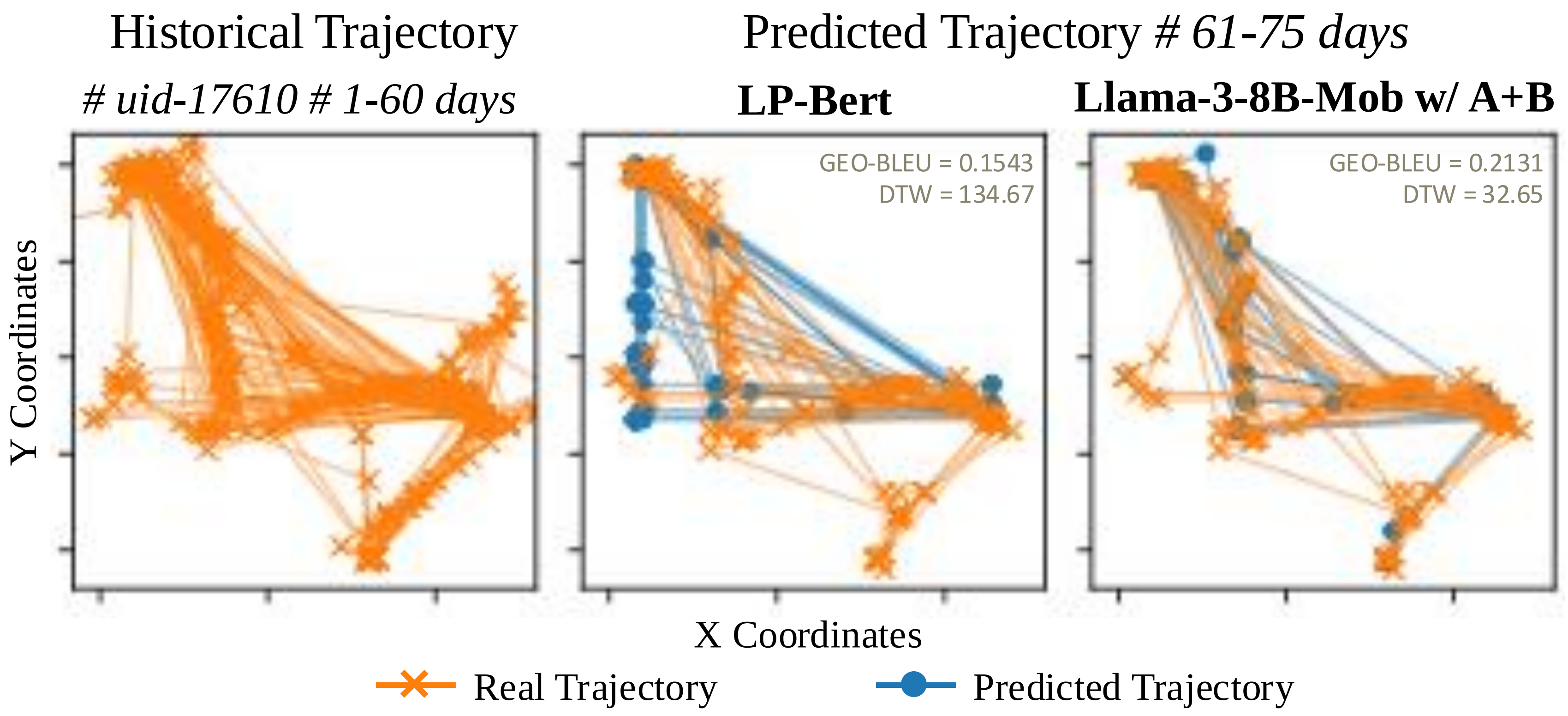
* All experiments were conducted on 4 NVIDIA RTX A6000 48GB GPUs.

- Training
  - ➢ Acceptable #Params after using LoRA.
  - ➢ Fine-tune time approaches 1 week !
- Inference
  - ➢ GPU Memory: 10x bigger than LP-Bert
  - ➢ Time: 16,000x slower than LP-Bert
    - Difficult to put into practice.
    - Increases linearly with trajectory length.



IV. Experiments

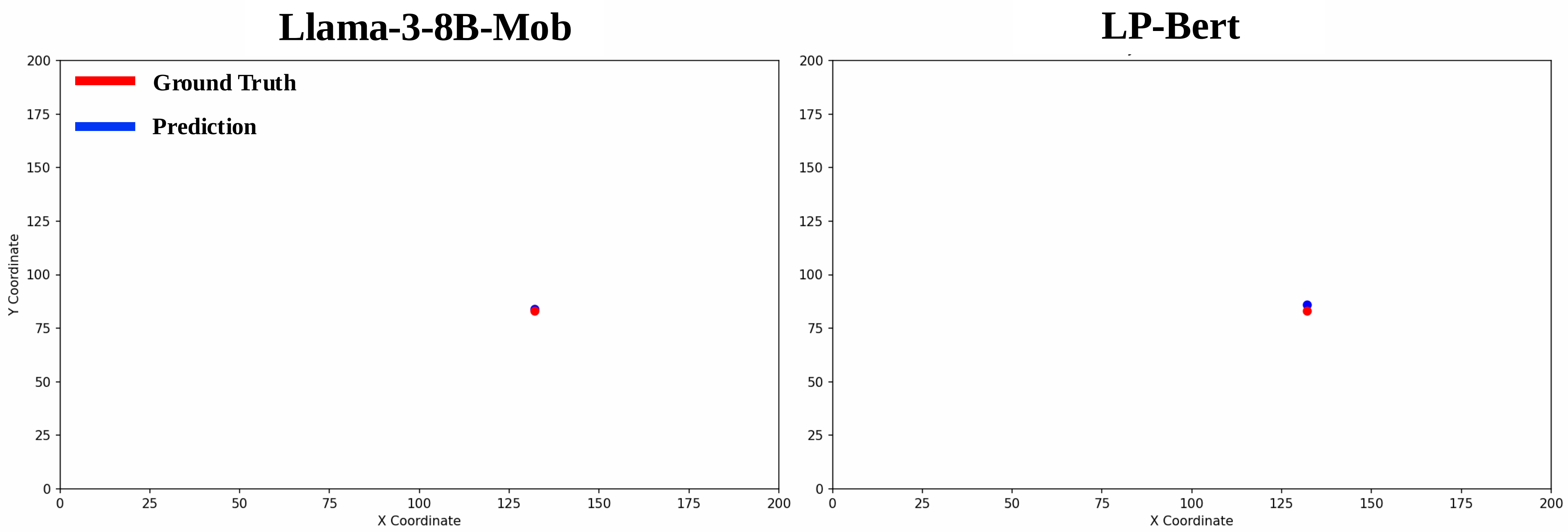
➢ Results -> Case Study



- LP-Bert tend to draw regular right triangles.
- Llama-3-8B-Mob could mimic human mobility very well.

IV. Experiments

➢ Results -> Case Study



- LP-Bert tend to draw regular right triangles.
- Llama-3-8B-Mob could mimic human mobility very well.

V. Summary & Take Away

➢ The Great Potential of LLMs in Trajectory Prediction

- **Instruction tuning Llama-3-8B** with only a **small amount of data** surpassing the SOTA and beat over 100+ teams in the HuMob'24.

- **Strong zero-shot generalizability** in learning human mobility patterns.
  - Effectively **generalizing to other cities from a single city data.**
  - Even when finetuned only **on limited samples**.

- **Larger models** may offer even **better capabilities**.
  - Llama-3-8B is just a **small model in the LLMs family**.

- **Improved instruction design** may enhance model performance.
  - Due to time limitations, there has not been an **exploration of different instructions**.
  - e.g., Chain of Thought.

V. Summary & Take Away

➢ Possible Future Directions

■ Improving Data Quality

- Data quality could be critical for effective fine-tuning.
- Trajectory data that better reflects the universal human behavior could greatly reduce training costs.

Model	Average DTW (↓)			Average GEO-BLEU (↑)			Mean Rank	# Trajs Used for Training
	B	*C*	*D*	*B*	*C*	*D*		
LP-Bert [13]	**23.30**	23.81	38.89	0.3093	0.2682	0.3033	4.17	113,600
Llama-3-8B-Mob *w/ B*	26.32	22.49	34.41	0.3322	0.2895	0.3157	2.50	17,600
Llama-3-8B-Mob *w/ C*	31.58	23.75	34.49	0.3399	0.2891	0.2833	3.67	13,600
Llama-3-8B-Mob *w/ D*	28.75	22.20	38.46	0.3251	0.2765	0.3056	3.50	2,400
Llama-3-8B-Mob *w/ A+B*	25.39	**20.57**	**31.94**	**0.3541**	**0.2969**	**0.3217**	**1.17**	18,600

■ Increasing Computational Efficiency.

- Current inference speed is too slow for practical applications (5mins per single user).
- Balancing between computational efficiency & prediction accuracy could be a crucial direction.

Thanks for your attention!
Q&A

Try Llama-3B-Mob

Contact us:
chuangyang@g.ecc.u-tokyo.ac.jp