Modeling Spatio-Temporal Trajectories with LLMs

PLM4Traj: Leveraging Pre-trained Language Models for Cognizing Movement Patterns and Travel Purposes from Trajectories

Yan Lin

lyan@cs.aau.dk
Data Engineering, Science and Systems
Aalborg University

September 1, 2024

AALBORG UNIVERSITY

Background

Spatio-temporal (ST) trajectory

- A sequence of timestamped locations
- Tracks the movements of an individual or object in a geographical space
- Enable various tasks and applications, such as trajectory prediction, anomaly detection, and trajectory user linking

Figure 1: A spatio-temporal trajectory.

Key of Utilizing ST Trajectories

Extraction of information

- **Movement patterns**: How the individual or object moves from one location to another
- **Travel purposes**: Underlying reason or motivation for the movement

Adaptability to tasks

Accurately perform a variety of downstream tasks, reducing the need for designing a separate method for each task.

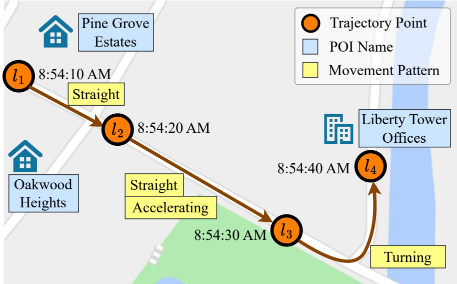


Figure 2: A trajectory of commuting to work.

Effectiveness of methods limited by capacities and scale and quality of available trajectory data.

Migrating LLMs to Trajectory Learning

Versatility of Large Language Models (LLMs)

- Promising results on various NLP tasks
- Benefit from their capacities and abundant large-scale corpus datasets

Similarities between trajectories and sentences in NLP

- ST correlations between trajectory points
- Movement patterns in trajectories
- Travel purposes of trajectories

- Contextual correlations between words
- Semantics of words
- Semantics of sentences

There is a significant potential in building a more effective trajectory learning model by leveraging LLMs.

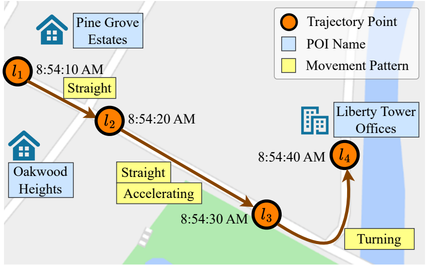
Challenges

LLMs are incapable of processing spatio-temporal features

- LLMs are designed to handle sequences of discrete word tokens as input
- Trajectories contain spatio-temporal features such as coordinates, timestamps, and road segments

LLMs are unable to extract movement patterns and travel purposes directly

- Movement patterns can be derived from changes in spatio-temporal features
- Travel purposes are closely linked to origin and destination locations



Some efforts are needed to migrate LLMs to trajectories.

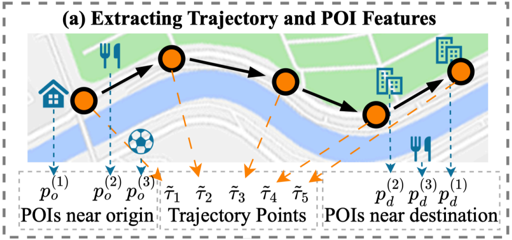
Trajectory and POI Feature Extraction

Trajectory Features

- Map each trajectory point τ_i onto the road network by Leuven Map Matching (LMM) algorithm
- Calculate the velocity v_i, acceleration a_i, and direction θ_i of each trajectory point

POI Features

- Retrieve the closest N_{POI} POIs to origin and destination
- Extract addresses and names of the retrieved POIs



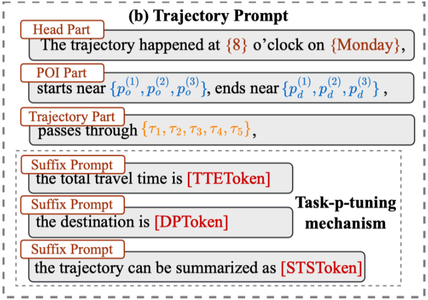
Trajectory Prompt Construction

Head Part, POI Part, and Trajectory Part

- ⟨Head Part⟩ enriches the input context and guides the LLM in analyzing trajectories
- ⟨POI Part⟩ provides information about POIs around the OD points
- ⟨Trajectory Part⟩ comprises the extracted features of the trajectory points

Suffix Part

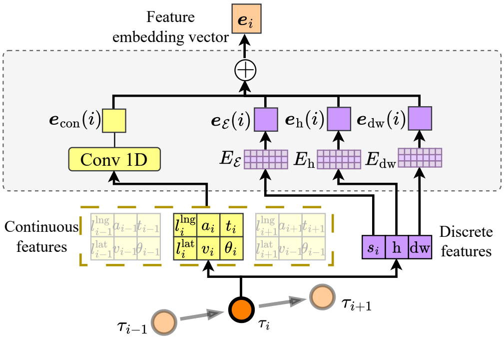
- A hybrid of hard and soft components
- The hard component signifies particular task
- The soft component [Token] is a task-specific token with a learnable embedding vector



Trajectory Prompt Embedding

Spatio-temporal Feature Embedding

Index-fetching embedding for discrete features and one-dimensional convolution for continuous features.



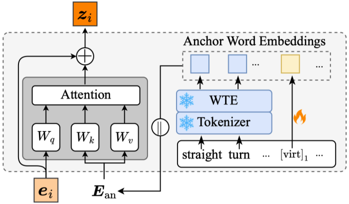
Trajectory Prompt Embedding

Spatio-temporal Feature Embedding

Index-fetching embedding for discrete features and one-dimensional convolution for continuous features.

Movement Pattern Semantic Projection

Employ a multi-head attention to project each ST feature embedding onto a semantic-rich textual space.



Categories	Words
Driving Behaviors	straight, turn, u-turn, brake, accelerate, decelerate, stop, overtake, zigzag, swerve, detour, slide, cruise, glide, cautious, reckless, leisurely
Traveling Dynamics	steady, smooth, rough, constant, dynamic, fast, slow, rapid, rushed, erratic, agile, stationary, sluggish

Trajectory Prompt Embedding

Spatio-temporal Feature Embedding

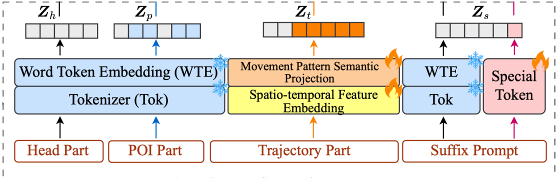
Index-fetching embedding for discrete features and one-dimensional convolution for continuous features.

Movement Pattern Semantic Projection

Employ a multi-head attention to project each ST feature embedding onto a semantic-rich textual space.

POI Feature Embedding

Embed POI features with LLM tokenizer and word token embedding.



(c) Trajectory Semantic Embedder

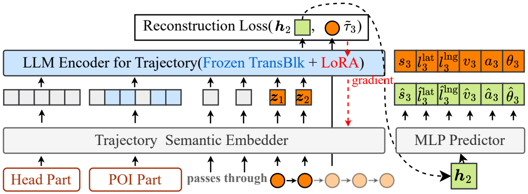
Model Training and Task Adaptation

Cross-reconstrution Pretext Task

- Autoregressively reconstruct the trajectory point features given ⟨POI Part⟩
- Autoregressively reconstruct each POI given ⟨Trajectory Part⟩

Task-specific Fine-tuning

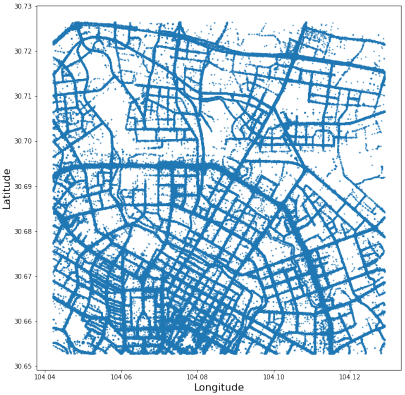
The proposed model is fine-tuned with the task's supervision to further improve prediction accuracy



Settings

Dataset

Dataset	Chengdu	Xi'an
Time span	09/30 - 10/10, 2018	09/29 - 10/15, 2018
#Segments	4,315	3,392
#Trajectories	140,000	210,000
#Records	18,832,411	18,267,440



Other Data source

- **Road network**: Openstreetmap
- **Map matching algorithm**: Leuven Map Matching
- **POIs**: Amap APIs

Performance Comparison

Overall Performance

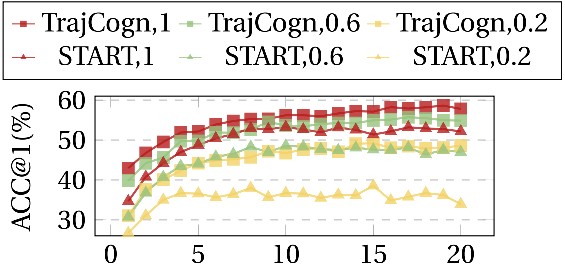
Our proposed method consistently outperforms the others and performs well across tasks.

Task		Travel Time Estimation			Destination Prediction			Similar Trajectory Search		
Datasets	Methods	RMSE (sec) ↓	MAE (sec) ↓	MAPE (%) ↓	ACC@1 (%) ↑	ACC@5 (%) ↑	Recall (%) ↑	Mean Rank ↓	ACC@1 (%) ↑	ACC@5 (%) ↑
Chengdu	Traj2vec	130.872 ± 2.013	59.993 ± 2.225	14.870 ± 0.698	43.074 ± 1.255	73.899 ± 1.568	14.760 ± 0.345	3.371 ± 0.156	83.325 ± 0.754	89.375 ± 0.459
	T2vec	128.508 ± 2.600	60.520 ± 2.575	15.224 ± 0.446	47.739 ± 0.239	73.509 ± 0.147	16.638 ± 0.108	3.345 ± 0.380	81.450 ± 0.778	93.700 ± 1.838
	TremBR	125.535 ± 2.849	57.965 ± 2.588	13.964 ± 0.860	48.987 ± 0.377	72.082 ± 0.289	17.010 ± 0.495	4.659 ± 1.010	83.980 ± 1.145	89.880 ± 0.303
	CTLE	132.636 ± 3.973	57.481 ± 1.144	13.153 ± 0.750	51.004 ± 0.683	79.434 ± 0.641	21.467 ± 0.704	9.429 ± 1.587	53.767 ± 7.414	69.200 ± 4.508
	Toast	128.793 ± 2.566	60.997 ± 3.537	14.883 ± 0.576	50.897 ± 0.495	79.664 ± 0.498	21.068 ± 0.383	5.944 ± 1.130	53.640 ± 2.244	71.600 ± 2.819
	TrajCL	120.211 ± 1.040	59.816 ± 1.841	14.741 ± 0.443	50.847 ± 0.249	79.693 ± 0.577	21.572 ± 0.324	1.198 ± 0.219	95.125 ± 5.022	98.875 ± 1.350
	START	122.205 ± 3.181	55.922 ± 2.397	12.717 ± 0.788	52.775 ± 0.311	80.423 ± 0.409	23.316 ± 0.310	1.089 ± 0.041	96.933 ± 2.060	99.900 ± 0.100
	LightPath	119.23 ± 2.367	55.614 ± 1.518	12.760 ± 0.854	49.154 ± 0.234	78.587 ± 0.583	20.660 ± 0.273	27.266 ± 3.544	74.267 ± 4.765	86.100 ± 3.874
	TrajCogn (ours)	**115.079 ± 1.608**	**51.973 ± 1.922**	**11.635 ± 0.587**	**59.594 ± 0.867**	**86.740 ± 0.294**	**30.184 ± 0.875**	**1.068 ± 0.044**	**99.240 ± 0.152**	**99.940 ± 0.060**
Xi'an	Traj2vec	187.010 ± 1.100	86.450 ± 2.884	13.634 ± 0.651	42.506 ± 0.394	75.761 ± 0.506	13.961 ± 0.376	2.284 ± 0.359	90.600 ± 0.704	98.017 ± 0.523
	T2vec	199.132 ± 2.447	86.008 ± 2.827	14.222 ± 0.495	43.596 ± 0.133	74.670 ± 0.343	13.527 ± 0.103	1.600 ± 0.340	89.467 ± 3.556	97.100 ± 1.637
	TremBR	185.727 ± 3.563	81.119 ± 2.411	12.770 ± 0.766	44.500 ± 0.349	75.111 ± 0.667	12.903 ± 0.741	3.478 ± 0.959	88.000 ± 1.355	93.000 ± 0.639
	CTLE	182.278 ± 2.665	79.712 ± 1.621	12.780 ± 0.571	44.837 ± 0.720	76.777 ± 0.610	14.826 ± 0.408	6.045 ± 1.149	41.200 ± 3.832	59.800 ± 9.835
	Toast	183.092 ± 3.827	84.925 ± 2.472	13.436 ± 0.627	45.078 ± 0.517	77.651 ± 0.123	15.459 ± 0.547	6.176 ± 1.042	30.600 ± 5.597	64.300 ± 6.505
	TrajCL	179.806 ± 3.298	82.494 ± 2.909	13.231 ± 0.270	45.807 ± 0.474	79.063 ± 0.596	16.836 ± 0.884	1.091 ± 0.024	95.625 ± 1.212	99.200 ± 0.116
	START	182.346 ± 3.254	80.763 ± 2.756	12.547 ± 0.501	46.127 ± 0.267	79.335 ± 0.489	16.306 ± 1.359	1.139 ± 0.201	95.925 ± 3.877	99.525 ± 0.763
	LightPath	180.032 ± 2.367	80.420 ± 2.189	12.253 ± 0.686	44.390 ± 0.247	72.753 ± 0.466	14.416 ± 0.539	13.877 ± 1.231	79.625 ± 3.236	91.700 ± 3.135
	TrajCogn (ours)	**166.884 ± 1.843**	**77.285 ± 2.086**	**11.357 ± 0.317**	**49.192 ± 0.238**	**81.763 ± 1.246**	**20.753 ± 0.210**	**1.083 ± 0.012**	**99.400 ± 0.254**	**99.800 ± 0.152**

Performance Comparison

Scalability Study

Our model demonstrates faster progress and achieves superior performance with less data compared to START.



Efficiency Study

While incorporating PLMs increases the model scale and reduces training speed, the additional learnable parameters and training speed remain reasonable.

Methods	Learnable Param (MB)	Pre-Train Speed (min/epoch)	Fine-Tune Speed (min/epoch)	Embed Time (sec)
CTLE	3.756	4.533	3.516	14.581
Toast	4.007	4.400	3.517	14.539
TrajCL	5.634	7.699	4.543	10.253
START	15.928	15.927	7.573	28.704
TrajCogn	27.922	24.931	19.644	110.516

Model Analysis

Ablation Study

The performance of the variants demonstrate the effectiveness of the proposed components.

Task	Travel Time Estimation			Destination Prediction			Similar Trajectory Search		
Methods	RMSE (sec) ↓	MAE (sec) ↓	MAPE (%) ↓	ACC@1 (%) ↑	ACC@5 (%) ↑	Recall (%) ↑	Mean Rank ↓	ACC@1 (%) ↑	ACC@5 (%) ↑
w/o PT	120.737 ± 0.634	54.951 ± 2.632	12.087 ± 0.980	57.455 ± 0.723	85.331 ± 0.161	28.390 ± 1.512	3.914 ± 0.033	88.000 ± 0.566	94.600 ± 0.707
w/o POI	116.132 ± 2.131	52.941 ± 4.453	12.080 ± 0.924	58.711 ± 0.215	86.128 ± 0.118	29.372 ± 0.666	1.092 ± 0.065	98.200 ± 2.115	99.325 ± 0.754
w/o Conv	117.038 ± 2.237	53.402 ± 3.175	11.836 ± 1.175	59.078 ± 1.054	86.200 ± 0.673	29.521 ± 1.477	1.137 ± 0.050	96.733 ± 1.823	98.700 ± 0.781
w/o PSP	115.454 ± 5.551	53.003 ± 2.363	12.265 ± 0.856	58.797 ± 0.698	86.166 ± 0.460	29.503 ± 0.779	1.256 ± 0.256	96.667 ± 2.214	98.367 ± 1.037
w/o $\mathscr{M}$	115.233 ± 0.509	52.790 ± 3.297	11.891 ± 0.794	58.930 ± 0.220	86.668 ± 0.324	29.626 ± 0.287	1.069 ± 0.022	98.525 ± 0.551	99.350 ± 0.100
TrajCogn (full)	**115.079 ± 1.608**	**51.973 ± 1.922**	**11.635 ± 0.587**	**59.594 ± 0.867**	**86.740 ± 0.294**	**30.184 ± 0.875**	**1.068 ± 0.044**	**99.240 ± 0.152**	**99.940 ± 0.060**

Model Analysis

Word Selection

Reducing vocabulary and replacing it with irrelevant words both lead to worse results, proving the rationality of our selection strategy.

Variants	ACC@1 (%)	ACC@5 (%)	Recall (%)
w/o $\mathcal{M}$	58.930 ± 0.220	86.668 ± 0.324	29.626 ± 0.287
Decrease	59.191 ± 0.291	86.791 ± 0.424	29.776 ± 0.439
Replace	58.107 ± 0.329	85.948 ± 0.237	28.798 ± 0.697
TrajCogn	**59.594** ± **0.867**	**86.740** ± **0.294**	**30.184** ± **0.875**

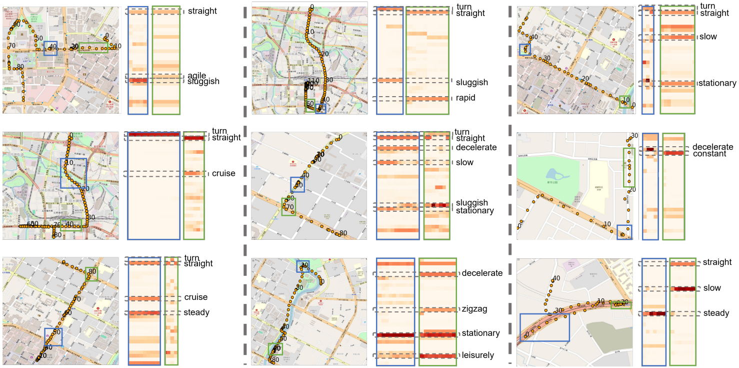
Additional Features

Downstream Task	Destination Prediction			Similar Trajectory Search		
Methods	ACC@1 (%) ↑	ACC@5 (%) ↑	Recall (%) ↑	Mean Rank ↓	ACC@1 (%) ↑	ACC@5(%) ↑
START	52.775 ± 0.311	80.423 ± 0.409	23.316 ± 0.310	1.089 ± 0.041	96.933 ± 2.060	<u>99.900 ± 0.100</u>
START w/ AF	53.287 ± 0.172	81.897 ± 0.191	23.897 ± 0.321	1.073 ± 0.006	96.200 ± 0.707	99.850 ± 0.071
TrajCogn w/o AF	<u>56.565 ± 0.360</u>	<u>85.023 ± 0.176</u>	<u>27.833 ± 0.302</u>	<u>1.072 ± 0.035</u>	<u>98.600 ± 1.097</u>	99.650 ± 0.336
TrajCogn	**59.594** ± **0.867**	**86.740** ± **0.294**	**30.184** ± **0.875**	**1.068** ± **0.044**	**99.240** ± **0.152**	**99.940** ± **0.060**

Case Study

Attention Map Visualization

Specific movement patterns displayed by trajectory points are associated with particular anchor words.



Zero-shot tasks

- LLMs are known for their ability to perform various NLP tasks without prior training.
- Execute different downstream tasks without fine-tuning?

Textual output

- The primary function of LLMs is generating textual content.
- Generate trajectory-related texts to enhance the models utilization in real-world applications?

Efficiency improvement

- PLM4Trajs size is relatively large for a trajectory learning model.
- Reduce the model size to improve computational and storage efficiency without sacrificing performance?

Contributors

Zeyu Zhou[1], Yan Lin[1], Haomin Wen, Shengnan Guo, Jilin Hu[2], Youfang Lin

School of Computer and Information Technology
Beijing Jiaotong University

[1] Both authors contributed equally to this research.
[2] School of Data Science and Engineering, East China Normal University

Thank you!

Yan Lin

lyan@cs.aau.dk

www.yanlincs.com