



北京工业大学
BEIJING UNIVERSITY OF TECHNOLOGY

ProDiff: Prototype-Guided Diffusion for Minimal Information Trajectory Imputation

ProDiff: 用于最小信息轨迹插补的原型引导扩散模型

汇报人：闫林枝

CONTENT_

01 背景_

02 方法_

03 实验_

04 结论_

P

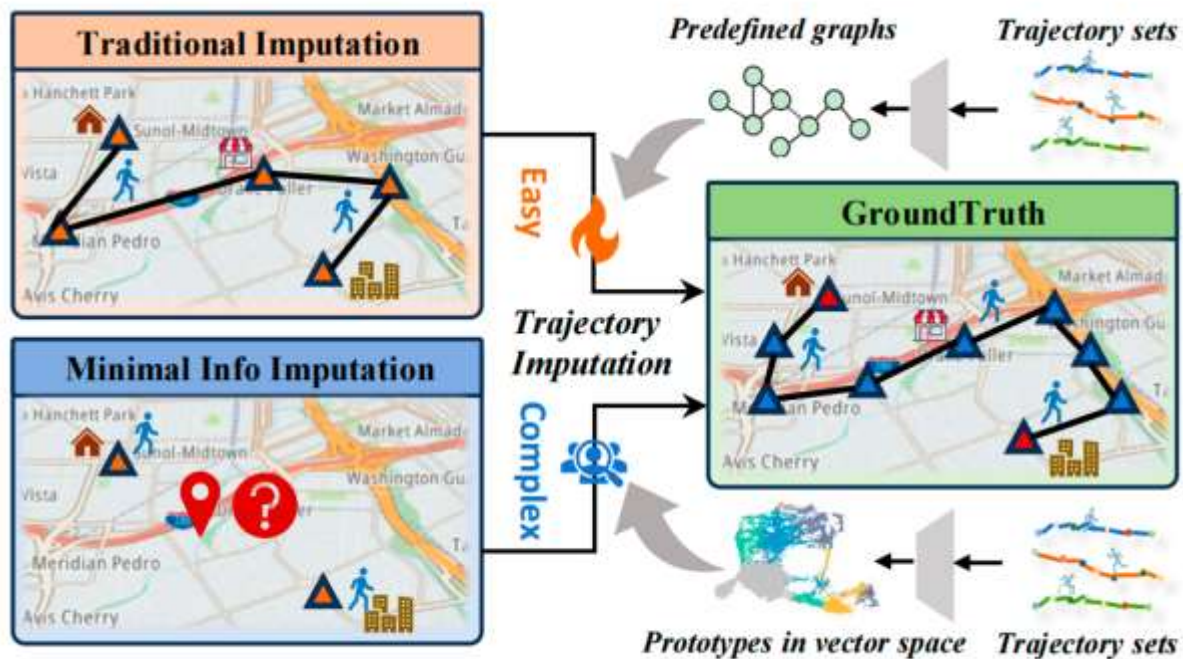
ART ONE

背景

01

背景与挑战

- **应用**：用于传染病控制、人类行为分析、城市规划等；
- **问题**：数据常不完整，因设备限制和采集场景多样；
- **传统方法局限**：数据依赖性强；受限于图结构的人工定义；对数据采集要求高。



预备知识

- **问题定义：** 给定轨迹 $\mathbf{X}_i = [\mathbf{x}_{i,1}, \dots, \mathbf{x}_{i,l}]$ ，其中每个点 $\mathbf{x}_{i,j} = \{t_{i,j}, lon_{i,j}, lat_{i,j}\}$ 表示时空坐标，将最小信息插补问题定义为仅给定端点 $\mathbf{X}_{i,1}$ 和 $\mathbf{X}_{i,l}$ 的情况下，重建 $\mathbf{X}_{i,2}, \dots, \mathbf{X}_{i,l-2}$.
- **轨迹序列窗口：** 为了处理轨迹，定义一个大小为 K 的滑动窗口，将轨迹划分为重叠的段。每个段表示为 $\mathbf{S}_p = [\mathbf{s}_{p,1}, \dots, \mathbf{s}_{p,k}]$ ，每个轨迹产生 $l - k + 1$ 个段， M 条轨迹就有 $\sum_{i=1}^M (l_i - k + 1)$ 段，其中长度小于 k 的轨迹被丢弃。
- **基础条件生成**
$$\mathbf{m}_j = \begin{cases} 1, & \text{if } j = 0 \text{ or } j = k, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

P

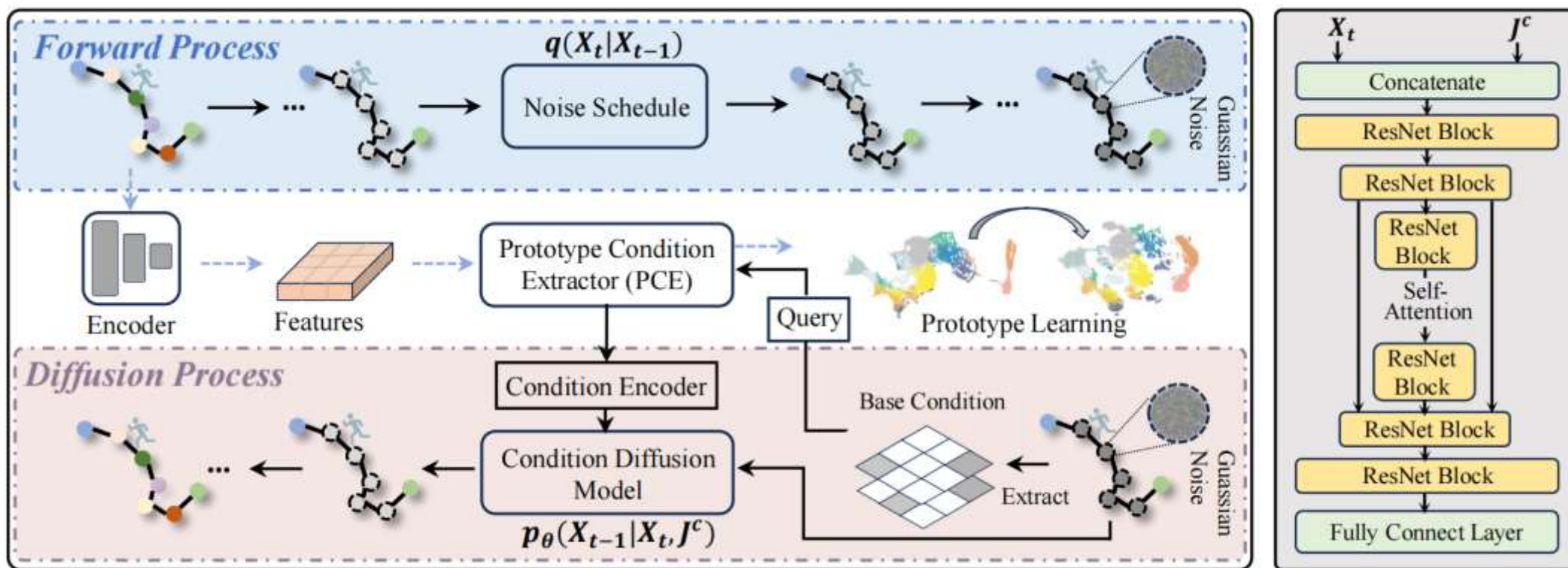
ART THREE

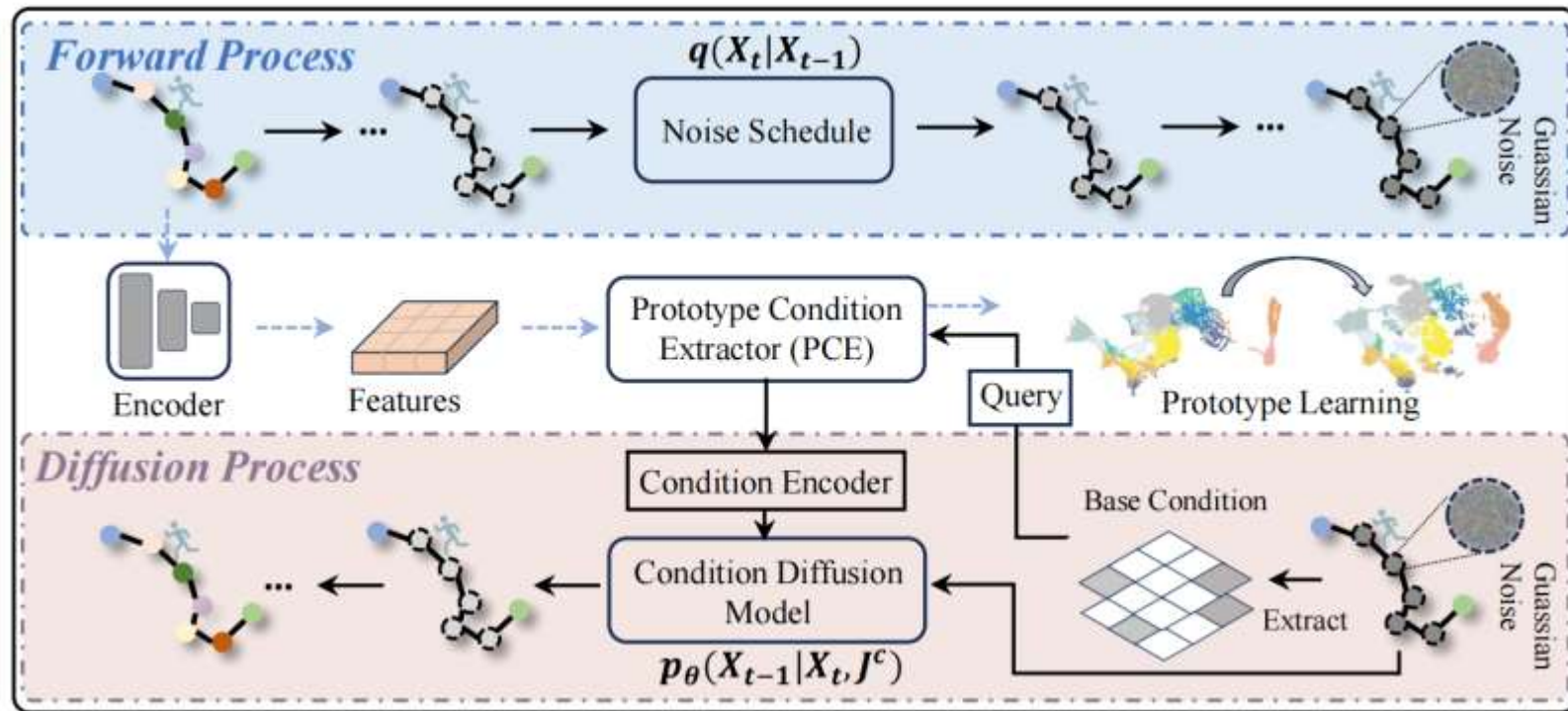
方法

02

Datasets

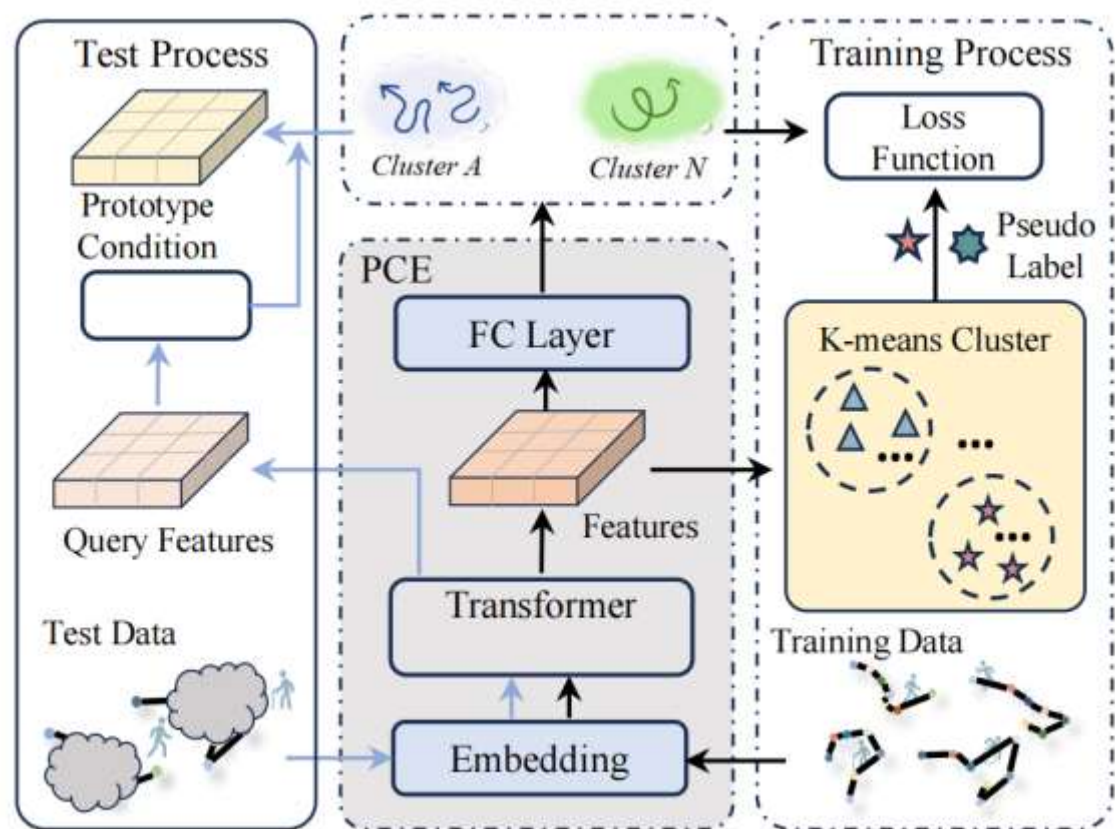
- 核心思路：仅用两个端点作为最小信息进行轨迹插补，整合原型学习和去噪扩散概率模型





- Forward Process（前向加噪）：通过 Noise Schedule 向轨迹添加高斯噪声，为后续“去噪重建”提供训练数据；
- Prototype Condition Extractor (PCE): 对特征进行聚类，生成原型（通勤、购物轨迹）；当需要插补轨迹时，用 Query 从原型中提取条件，引导扩散模型生成合理轨迹；
- Diffusion Process（扩散去噪 / 轨迹重建）：输入：带噪声的轨迹 X_t + 条件 J^c （包含原型条件 + 基础条件）
Condition Encoder 整合条件信息，生成指导去噪的信号，Condition Diffusion Model 用 1D - UNet 对带噪声轨迹逐步去噪，预测噪声并还原轨迹，实现“仅用端点 + 原型条件”插补轨迹。

基于原型的条件提取器



- **PCE:** 连接“轨迹数据”和“原型模式”的桥梁；
- **训练流程:** 从训练数据中学习“人类移动原型”，为插补任务提供先验模式；
- **测试流程:** 用学习到的原型，仅通过端点Query补全轨迹。

PCE

✓ 轨迹嵌入与原型生成

$$\mathbf{H}_i = \sum_j^k (\text{Encoder}(s_{i,j})). \quad (3)$$

- 编码器捕捉轨迹点的时空语义，整合轨迹的全局时空依赖

$$\mathbf{P} = \mathbf{W}_p \mathbf{H}_p + \mathbf{b}_p, \quad (4)$$

- 通过全连接层生成原型向量 \mathbf{P} ， \mathbf{P} 代表从大量轨迹中提炼出的“典型移动模式”（如“早高峰通勤”“周末购物”等原型）

✓ 无监督聚类与伪标签生成

- 对轨迹特征 \mathbf{H}_p 应用 K-means 聚类，划分为 N_c 个簇，生成伪标签 p_{kmeans}

✓ 为扩散模型提供条件

$$\mathbf{D} = [\text{Dis}(\mathbf{Q}_b, P_1), \dots, \text{Dis}(\mathbf{Q}_b, P_{N_p})], \quad (5)$$

$$\mathbf{P}^c = \mathbf{D}^T \mathbf{P}. \quad (6)$$

PCE

✓ 双损失优化原型空间

$$\mathcal{L}_{C1}(\gamma) = - \sum_{i=1}^{N_c} p_{kmeans}^i \log(q_{proto}^i), \quad (12)$$

分类一致性损失：通过交叉熵强制原型预测标签与 K-means 伪标签一致，确保原型学习 “对齐” 聚类规律。

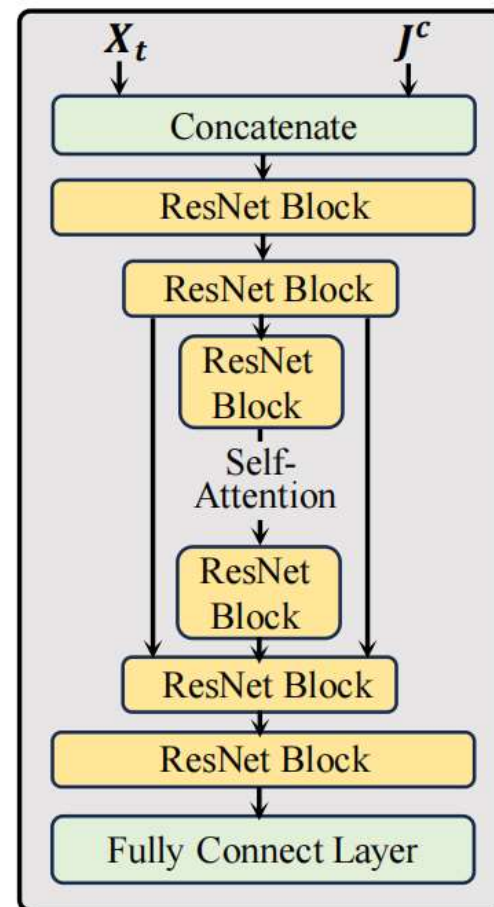
$$\mathcal{L}_{C2}(\gamma) = \mathbb{E}[\max(0, d(\mathbf{H}_i, \mathbf{P}^+) - d(\mathbf{H}_i, \mathbf{P}^-) + m)], \quad (13)$$

对比损失：使同类轨迹特征更接近、异类更远离，增强特征空间判别性。

✓ 优化原型空间的判别性，让同类轨迹特征更接近、不同类更远离。

基于扩散的生成模型

- 输入：带噪声的轨迹 X_t + 条件 J^c （包含原型条件 P^c + 基础条件 B^c ）；
- ResNet Block：提取局部时空特征，避免梯度消失；
- Self - Attention：捕捉长距离依赖（比如轨迹起点和终点的关联）。
- 通过 Fully Connect Layer 预测噪声，用于去噪重建。



基于扩散的生成模型

✓ 前向加噪声过程

$$d\mathbf{Z} = \mathbf{f}(\mathbf{Z}, t)dt + g(t)d\mathbf{w}, \quad (8)$$

✓ 去噪网络构建

✓ 条件融合与去噪

$$\mathcal{J}^c = WD(\mathbf{B}^c) + WD(\mathbf{P}^c). \quad (7)$$

基础条件仅约束“起点和终点”，原型条件补充“中间路径的模式先验”（如“通勤通常走主干道”）。同时满足“端点约束”和“模式合理性”。

✓ 噪声预测

$$\mathcal{L}_J(\theta, \gamma) = \mathbb{E}_{t \sim \mathcal{U}} \mathbb{E}_{\mathbf{Z}_0 \sim p, \epsilon \sim \mathcal{N}} [\|\epsilon - \epsilon_\theta(\mathbf{Z}_t, t, f_\gamma(\mathbf{Z}_0))\|^2]. \quad (11)$$

$$\mathcal{L}(\theta, \gamma) = \lambda_1 \mathcal{L}_J(\theta, \gamma) + \lambda_2 \mathcal{L}_{C1}(\gamma) + \lambda_3 \mathcal{L}_{C2}(\gamma), \quad (14) \quad \text{最终损失函数}$$

P

ART FOUR

实验

数据集

Table 6. Statistics of two human mobility datasets.

Dataset	WuXi	FourSquare
Time Span (day)	111	310
Used Time Span (day)	10	310
Train Active Users	30000	1834
Test Active Users	3000	459
Location Updates	671,124	573,703
Average Distance (meter)	3336.33	4301.51
Average Time (hour)	7.8	37.15

- 评价指标:

对于时间序列插值方法

轨迹覆盖率: 生成点 $s_{i,j}$ 中与真实值的距离在阈值 τ

$$TC@ \tau = \frac{1}{k} \sum_{j=1}^k \mathbb{I} \left(d \left(\widehat{s_{i,j}}, s_{i,j} \right) < \tau \right), \quad (15)$$

- 无锡: 用户ID、基站ID、时间戳以及一个标签
- 四方: 时间戳、全球定位系统 (GPS) 坐标以及语义信息

对于特定于轨迹的基线方法

- **密度**: 衡量真实轨迹与生成轨迹之间网格密度的余弦相似度 (数值越高越好)。
- **距离**: 评估真实数据与生成数据之间出行距离的差异, 通过计算连续点之间距离的总和得出 (数值越低越好)。
- **线段距离**: 评估真实数据与生成数据之间线段距离的差异, 以内的比例定义为连续点之间的距离 (数值越低越好)。
- **半径**: 评估所有活动位置与中心位置的均方根距离, 以此表明空间范围 (数值越低越好)。
- **MAE**: 衡量真实轨迹与生成轨迹之间误差的平均幅度 (数值越低越好)。
- **RMSE**: 评估预测值与实际值之间平均平方差的平方根 (数值越低越好)。

主要结果

- ✓ 与时间序列插值方法的对比，ProDiff在所有数据集上始终优于序列插补模型；
- ✓ 与轨迹特定方法的对比，ProDiff在空间分布指标（如密度、距离、分段距离、半径、平均绝对误差、均方根误差）上表现出显著优势。

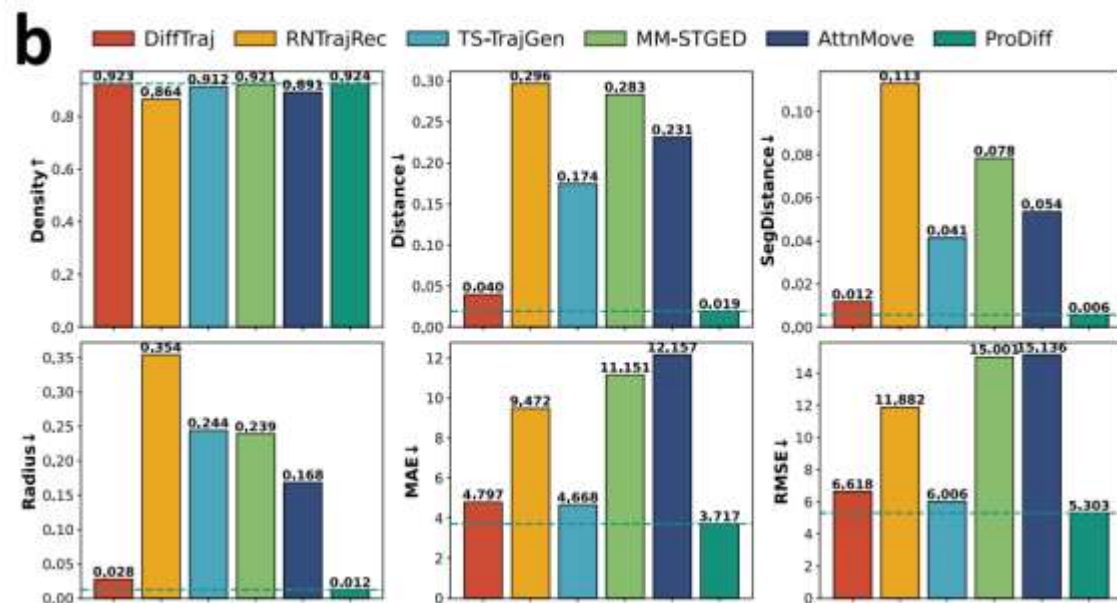
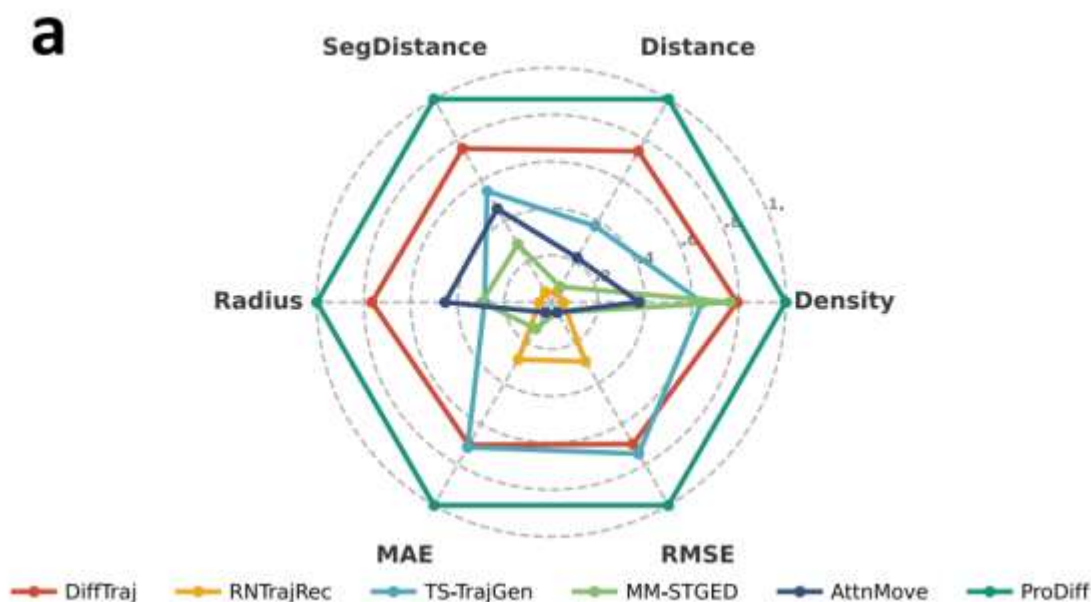


Table 1. Comparison of model performance for different thresholds and different trajectory lengths on WuXi and FourSquare.

Method	WuXi					FourSquare				
	TC@2k	TC@4k	TC@6k	TC@8k	TC@10k	TC@2k	TC@4k	TC@6k	TC@8k	TC@10k
VAR (Lütkepohl, 2013)	0.5194	0.5632	0.6050	0.6441	0.6811	0.5000	0.5000	0.5000	0.5000	0.5000
SAITS (Du et al., 2023)	0.5059	0.5224	0.5498	0.5861	0.6311	0.5000	0.5000	0.5000	0.5000	0.5000
TimesNet (Wu et al., 2023)	0.5080	0.5290	0.5593	0.5955	0.6352	0.5015	0.5054	0.5133	0.5258	0.5431
Diff-TS (Yuan & Qiao, 2024)	0.5123	0.5462	0.5951	0.6496	0.7060	0.5268	0.5714	0.6173	0.6571	0.6932
DiffTraj (Zhu et al., 2024a)	0.6958	0.8198	0.8816	0.9169	0.9402	0.5945	0.6845	0.7574	0.8189	0.8666
Diff+Mask (Ours)	0.6584	0.7731	0.8400	0.8834	0.9159	0.6541	0.7379	0.8010	0.8525	0.8928
ProDiff (Ours)	0.7155	0.8414	0.9006	0.9326	0.9520	0.6644	0.7452	0.8087	0.8596	0.8971
VAR (Lütkepohl, 2013)	0.3360	0.3437	0.3556	0.3692	0.3840	0.3333	0.3333	0.3334	0.3334	0.3335
SAITS (Du et al., 2023)	0.3427	0.3762	0.4275	0.4880	0.5533	0.3333	0.3333	0.3333	0.3333	0.3333
TimesNet (Wu et al., 2023)	0.3419	0.3654	0.4029	0.4500	0.5044	0.3386	0.3530	0.3756	0.4039	0.4341
Diff-TS (Yuan & Qiao, 2024)	0.3515	0.4011	0.4726	0.5491	0.6211	0.3761	0.4283	0.4827	0.5383	0.5874
DiffTraj (Zhu et al., 2024a)	0.5976	0.7476	0.8227	0.8688	0.9005	0.4277	0.5404	0.6428	0.7314	0.8025
Diff+Mask (Ours)	0.5767	0.7324	0.8228	0.8802	0.9180	0.4859	0.5970	0.6902	0.7671	0.8265
ProDiff (Ours)	0.5978	0.7686	0.8518	0.8992	0.9285	0.5005	0.6093	0.7013	0.7772	0.8345
VAR (Lütkepohl, 2013)	0.2537	0.2627	0.2739	0.2861	0.2986	0.2500	0.2500	0.2500	0.2500	0.2500
SAITS (Du et al., 2023)	0.2572	0.2764	0.3059	0.3485	0.3976	0.2500	0.2502	0.2505	0.2509	0.2513
TimesNet (Wu et al., 2023)	0.2520	0.2574	0.2663	0.2785	0.2942	0.2516	0.2563	0.2634	0.2715	0.2808
Diff-TS (Yuan & Qiao, 2024)	0.2689	0.3199	0.3907	0.4676	0.5453	0.3233	0.3932	0.4611	0.5358	0.5964
DiffTraj (Zhu et al., 2024a)	0.5418	0.7009	0.7868	0.8414	0.8795	0.3316	0.4526	0.5671	0.6688	0.7520
Diff+Mask (Ours)	0.4486	0.5946	0.6943	0.7631	0.8107	0.3957	0.5300	0.6431	0.7350	0.8045
ProDiff (Ours)	0.5752	0.7501	0.8236	0.8663	0.8945	0.4000	0.5331	0.6474	0.7404	0.8090
VAR (Lütkepohl, 2013)	0.2012	0.2047	0.2102	0.2177	0.2270	0.2000	0.2000	0.2000	0.2000	0.2000
SAITS (Du et al., 2023)	0.2080	0.2316	0.2686	0.3158	0.3692	0.2000	0.2000	0.2000	0.2000	0.2000
TimesNet (Wu et al., 2023)	0.2073	0.2275	0.2591	0.3035	0.3559	0.2003	0.2013	0.2034	0.2064	0.2110
Diff-TS (Yuan & Qiao, 2024)	0.2173	0.2655	0.3367	0.4190	0.5000	0.2751	0.3484	0.4207	0.4990	0.5646
DiffTraj (Zhu et al., 2024a)	0.4994	0.6687	0.7640	0.8259	0.8692	0.2762	0.4024	0.5300	0.6453	0.7386
Diff+Mask (Ours)	0.3793	0.5104	0.6046	0.6773	0.7344	0.3412	0.4800	0.5999	0.7023	0.7868
ProDiff (Ours)	0.4996	0.6994	0.8048	0.8667	0.9053	0.3522	0.4910	0.6105	0.7146	0.7920

消融实验

- ✓ **PCE**的影响：为评估各个模块的影响，移除了PCE、交叉熵损失以及对比损失，同时保持联合损失不变；
- ✓ **跨生成模型的泛化能力**：为评估PCE的泛化能力，我们将其与条件变分自编码器（cVAE）和条件生成对抗网络（cGAN）相结合，对这些模型同时应用掩码（MASK）和PCE。

Table 3. Performance comparison of removing different modules.

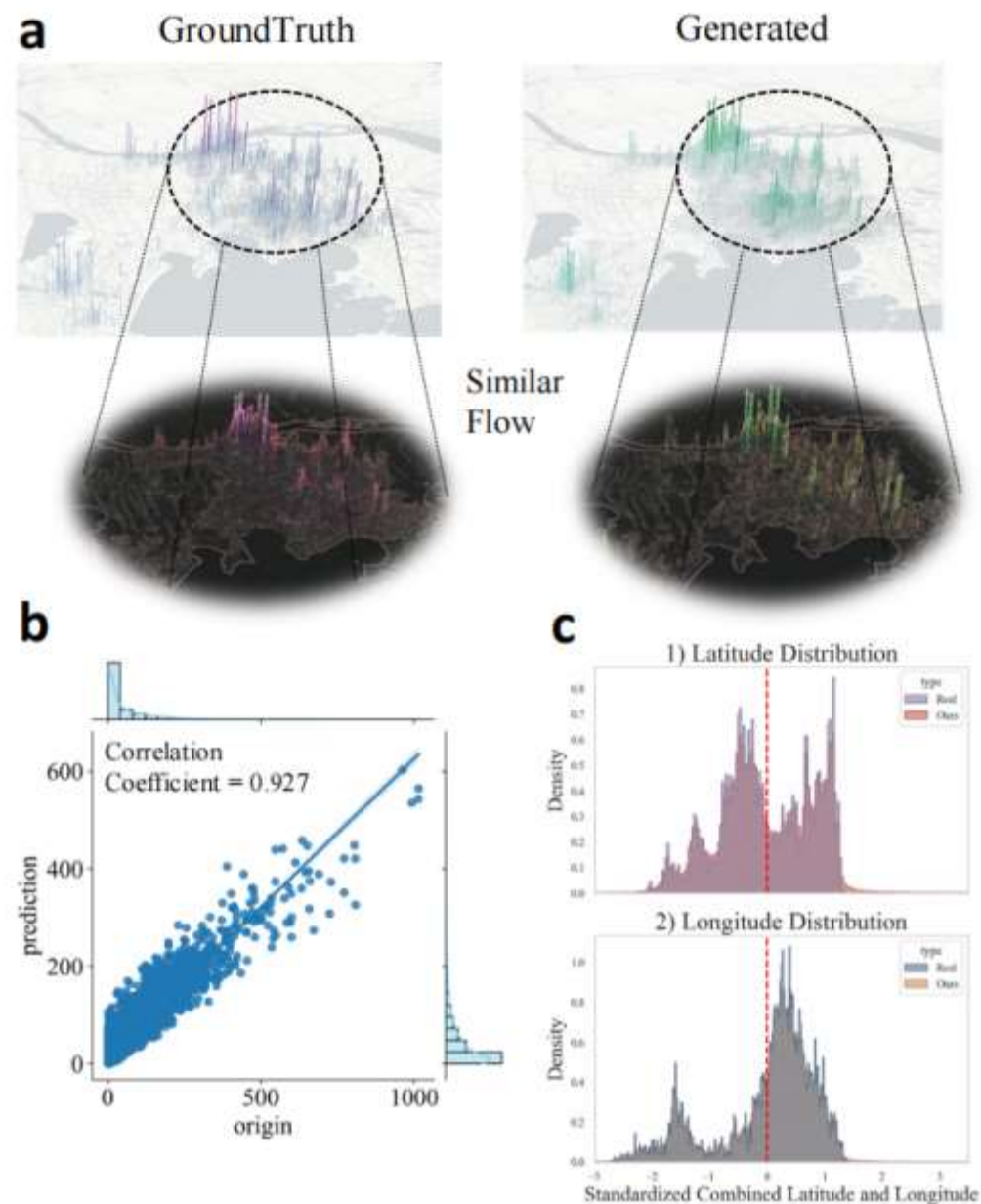
	Method	TC@2k	TC@4k	TC@6k	TC@8k	TC@10k
$K=6$	ProDiff	0.5978	0.7686	0.8518	0.8992	0.9285
	w.o. Pro	0.5767	0.7324	0.8228	0.8802	0.9180
	w.o. \mathcal{L}_{C1}	0.5939	0.7556	0.8371	0.8867	0.9195
	w.o. \mathcal{L}_{C2}	0.5952	0.7560	0.8374	0.8869	0.9199
$K=8$	ProDiff	0.5752	0.7501	0.8236	0.8663	0.8945
	w.o. Pro	0.4486	0.5946	0.6943	0.7631	0.8107
	w.o. \mathcal{L}_{C1}	0.5395	0.7205	0.7966	0.8399	0.8691
	w.o. \mathcal{L}_{C2}	0.4888	0.6638	0.7473	0.7984	0.8340

Table 4. Performance comparison for cVAE and cGAN.

Method	TC@2k	TC@4k	TC@6k	TC@8k	TC@10k
cVAE+MASK	0.2616	0.2936	0.3385	0.3926	0.4513
cVAE+pro	0.3416	0.3685	0.4082	0.4540	0.5009
cGAN+MASK	0.2760	0.3240	0.3742	0.4285	0.4896
cGAN+pro	0.3074	0.3997	0.4746	0.5361	0.5883

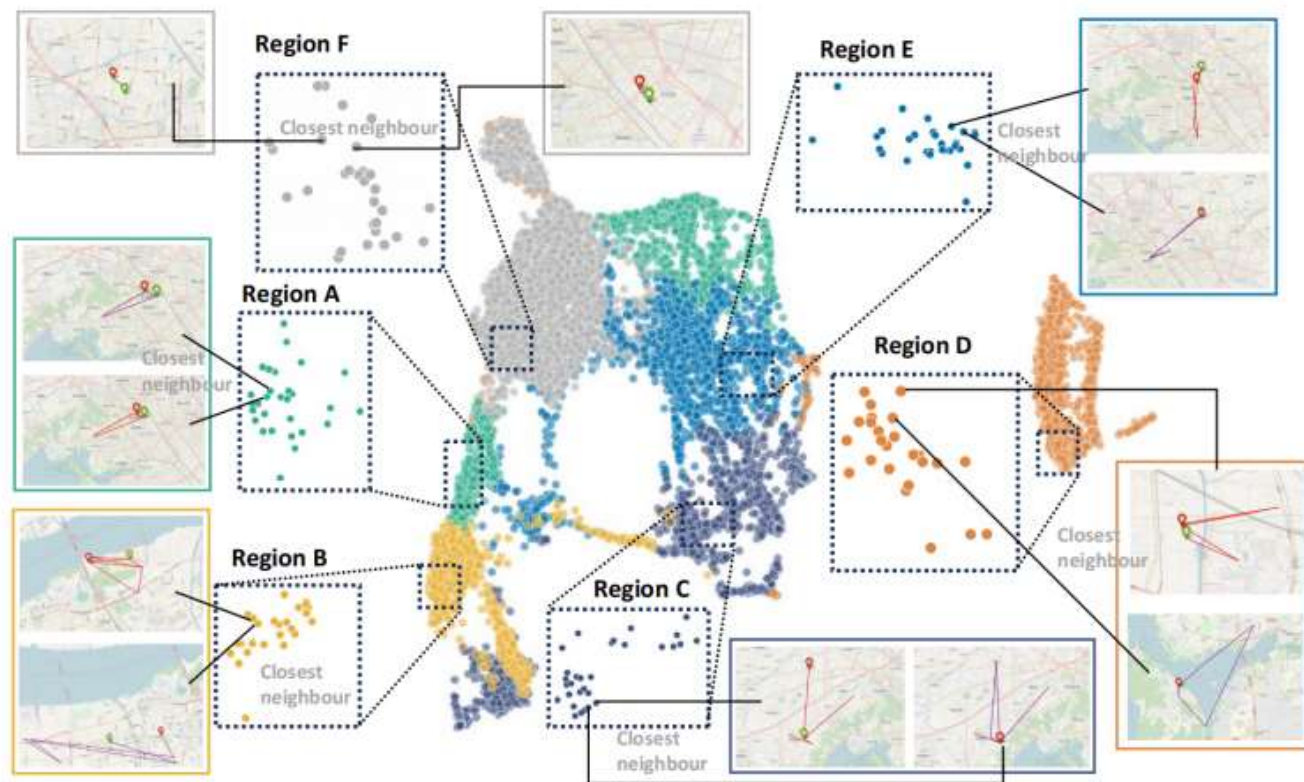
生成数据的实用性

- ✓ 为评估ProDiff生成数据在现实世界中的适用性，在无锡测试了其在**交通流量分析**方面的性能。
- ✓ 在无锡，利用针对7000多人在10天内的轨迹插补（其中 $k = 6$ ）进行了相关研究。将这座城市划分成了边长为1公里 \times 1公里的网格，每当个体的轨迹经过某个网格时，该网格的值就会相应增加。
- ✓ 结果表明：ProDiff能够生成逼真且可用的轨迹数据，使其适用于下游的人员流动分析任务。



可解释性

- 原型学习是否能捕捉到低维空间中可解释的运动模式？
- 可视化展示，轨迹数据被输入到经过训练的PCE中，使用K均值聚类（选取前6类），然后进行降维处理；
- 结果表明：该模型具备嵌入人类轨迹、捕捉结构化运动行为、区分差异以及在训练过程中优化表示的能力，并且在与扩散框架相结合时，能够提升轨迹插补的性能。



P

ART FIVE

结论

05

结论

- ✓ 解决轨迹插补问题，重点关注如何利用最少的信息生成逼真的轨迹；
- ✓ 提出了ProDiff，这是一种原型引导的扩散模型，它能捕捉宏观层面的移动模式，同时在轨迹生成过程中保持较高的保真度；
- ✓ 实验表明，ProDiff在两个数据集上的表现优于现有最先进的方法，提高了轨迹插补的准确性；
- ✓ 消融实验证实，原型学习显著增强了轨迹表示能力，而扩散建模则有效地重构了逼真的运动轨迹。

感谢观看！

汇报人：闫林枝

> GOODBYE <