MM-Path: Multi-modal, Multi-granularity Path Representation Learning—Extended Version

多模态、多粒度路径表示学习——扩展版本

2025/2/23

I Introduction

I Introduction

研究背景

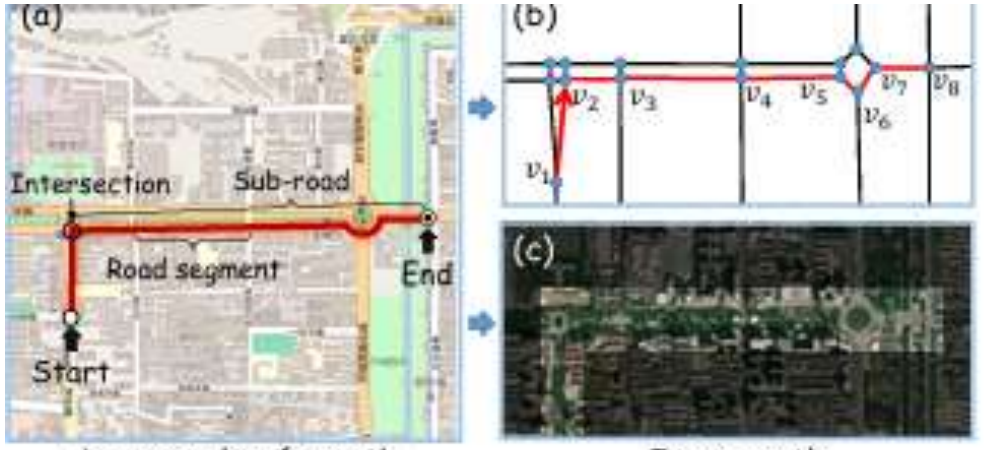
- 路径几乎影响着我们生活的方方面面。理解路径并开发有效的路径表示变得越来越重要，为多个不同领域提供了宝贵的洞察。
- 预训练的路径表示学习模型能够以无监督的方式高效地生成通用的路径表示。通过简单的微调和少量的标注数据，它们能够适应多种下游任务。

I Introduction

研究目的

路径拥有不同的模态表示方法，这些模态能够从不同方面提供更加丰富与不同的信息，例如：

- 道路路径：此路径从道路网络导出的路径（简称道路路径）阐明了路径中各路段之间的拓扑关系
- 图像路径：此路径从路径的遥感图像拍摄，提供了几何特征和更广泛的环境背景



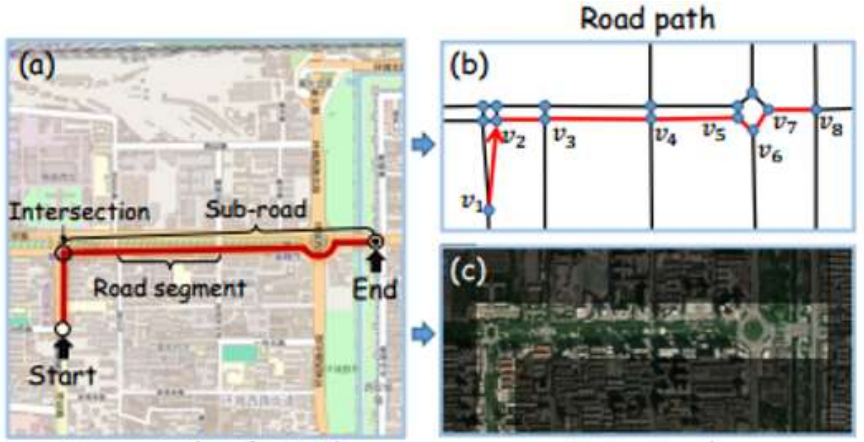
An example of a path　　Image path

然而，当前的路径表示学习模型主要依赖于来自道路网络的单模态数据，这无法捕捉到对路径全面理解所需的深层、全面的上下文信息。这促使我们开发一种多模态预训练的路径表示学习模型。

面临挑战

构建这样的模型面临两个主要个挑战：

道路路径和图像路径之间的信息粒度差异显著阻碍了跨模态语义对齐：道路路径和图像路径之间的信息粒度差异很大。道路路径通常关注详细的拓扑结构和道路连通性，而图像路径则在较大范围内捕捉全局环境背景，反映相应区域的功能属性。道路在本质上具有不同的粒度，包括路口、道路段和子道路。在不同粒度上充分理解路径可以提供从微观到宏观层面的见解，减轻模态间信息粒度差异所导致的负面影响. 虽然一些研究已经探索了单一模态数据的多粒度，但它们尚未充分解决多模态环境中多粒度分析的要求



An example of a path　　Image path

道路路径和图像路径固有的异质性在特征融合过程中构成了重大挑战：

道路路径和图像路径之间的数据结构和信息粒度差异延伸到它们的学习方法。

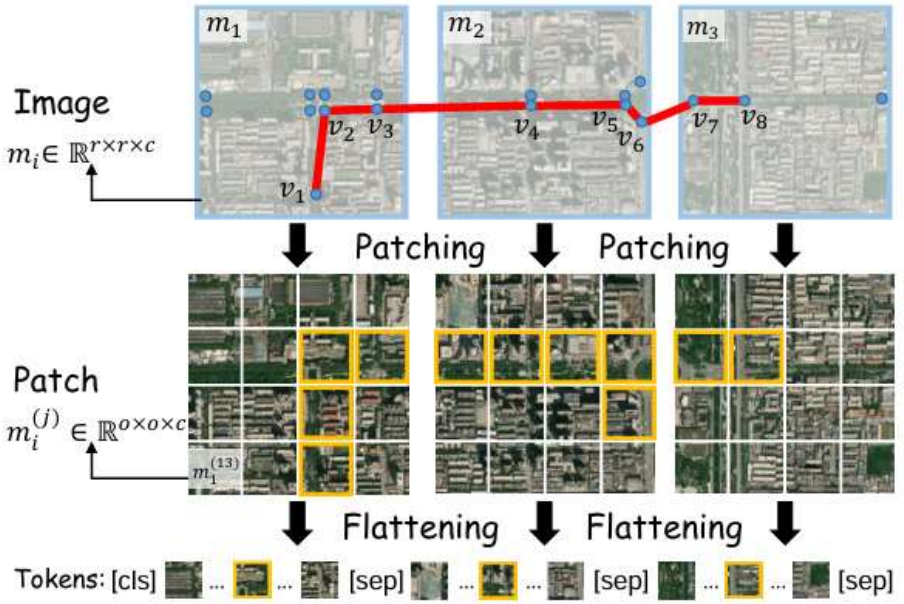
- 道路路径表示学习通常关注道路和路口之间的连通性和可达性，以及分析图结构
- 图像路径的图像学习方法优先进行对象识别和特征提取，旨在广泛理解图像内容。

II Methods

II- Basic Conception

路径:路径 p 是一系列连续的节点，可以从道路网络视图和图像视图中观察到。
道路网络:道路网络表示为 G = (V, E)，其中 V 和 E 分别表示一组节点和边。节点 $v \in$ V 是道路交叉口或道路终点。边 $e \in$ E 表示连接两个节点的道路段。
道路路径:我们定义道路网络上路径 p 的节点序列为道路路径 R (p) = $\langle v1, v2, ..., v|$ R (p) $|\rangle$，其中每个元素表示一个节点，|R (p)| 表示道路路径 R (p) 的长度。需要注意的是，道路路径中任何相邻节点之间必须有一条边 $e \in$ E 连接。

图像路径:给定一个感兴趣的区域，我们将该区域划分为固定大小的片段，生成一组由不重叠的固定大小遥感图像组成的图像集 Mrs。该集合中的每个图像表示为 $m \in$ R$r \times r \times c$，其中 c 表示通道数，(r, r) 表示分辨率。随后，给定道路路径 R (p)，图像路径（即路径的图像序列）M (p) 通过选择一系列图像 mi 形成，这些图像对应于道路路径中节点所在的特定纬度和经度。例如，如图 2 的上部所示，考虑道路路径 R (p) = $\langle v1, ..., v8\rangle$，其中节点 $v1$、$v2$ 和 $v3$ 位于图像 $m1$ 中，节点 $v4$、$v5$ 和 $v6$ 位于图像 $m2$ 中，节点 $v7$ 和 $v8$ 位于图像 $m3$ 中。这样就形成了图像路径 M (p) = $\langle m1, m2, m3\rangle$。



道路子路径:给定一条道路路径 R(p) 和一条图像路径 M(p)，位于相同图像中的 R(p) 的节点属于一条道路子路径。

目标:

$$\mathbf{x} = f(\mathcal{R}(p), \mathcal{M}(p))$$

II- 整体框架

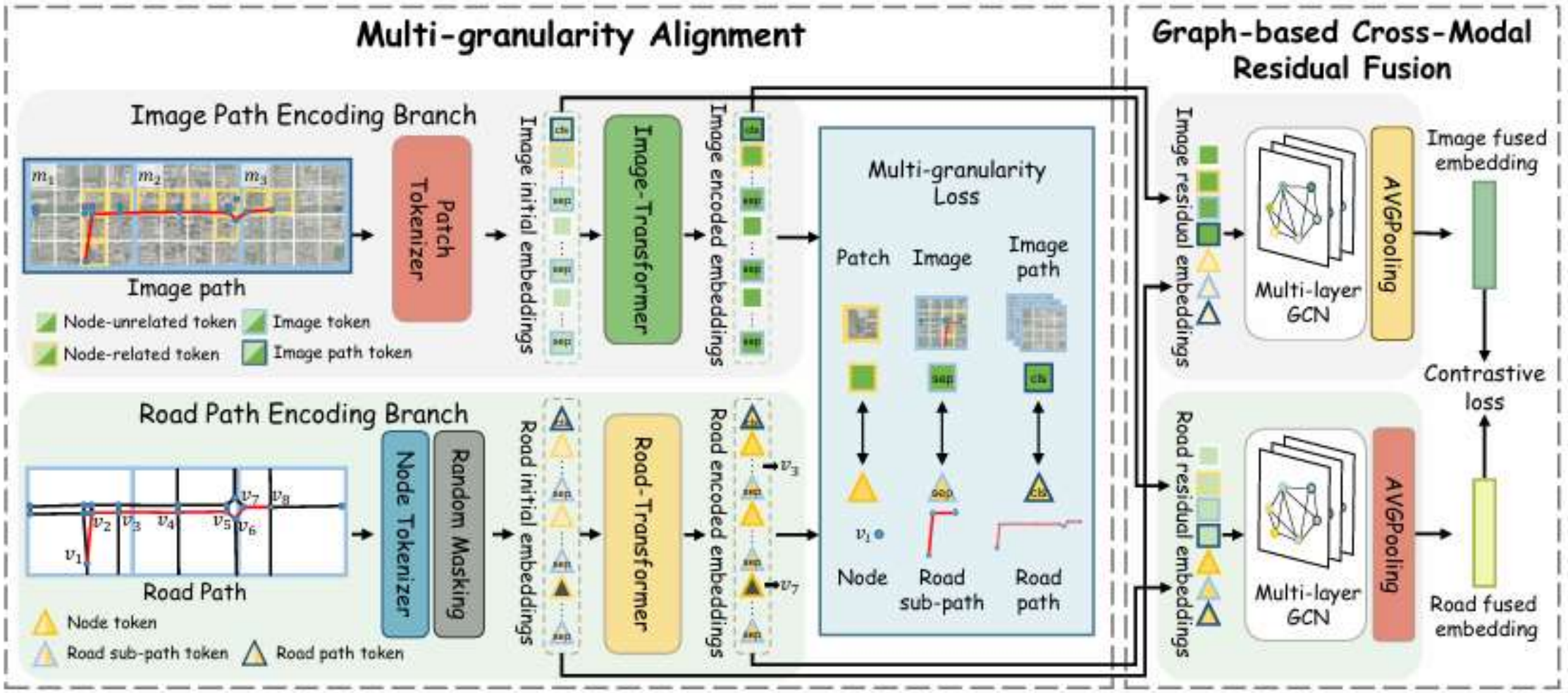


Figure 3: Overall framework of MM-Path

II-初始嵌入

- 分词器将图像路径中的每张图像分割成一系列小区域，以提取细粒度的语义信息
- 每个小区域 $mi^{(j)}$ 随后被转换成一个嵌入向量$\mathbf{mi}^{(j)}$，这可以使用预训练的ResNet50进行初始化。图像的初始嵌入通过将这些小区域的嵌入与图像位置嵌入相加来计算。

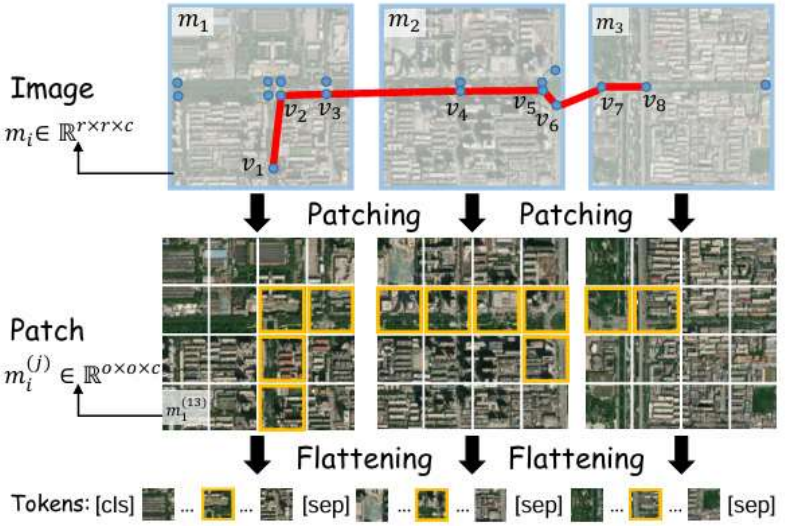
$$\mathbf{H}^{(0)} = [\mathbf{m}_{\text{cls}}, \mathbf{m}_1^{(1)}, \ldots, \mathbf{m}_{|\mathcal{M}(p)|}^{(r^2/o^2)}, \mathbf{m}_{\text{sep}}] + \mathbf{T}_{\text{image}}$$

- 道路路径的建模与图像类似。例如，节点标记序列是

$$[\text{cls}, v_1, v_2, v_3, \text{sep}, v_4, v_5, v_6, \text{sep}, v_7, v_8, \text{sep}]$$

- 道路初始嵌入

$$\mathbf{P}^{(0)} = [\mathbf{v}_{\text{cls}}, \mathbf{v}_1, \ldots, \mathbf{v}_{|\mathcal{R}(p)|}, \mathbf{v}_{\text{sep}}] + \mathbf{T}_{\text{road}}$$



II-编码嵌入

- 建立道路路径和图像路径编码分支来处理这两种模态。这些初始嵌入随后通过道路和图像转换器分别处理，以产生道路和图像编码嵌入

$$\mathbf{H}^{(j)} = \text{Image-Transformer}(\mathbf{H}^{(j-1)}),$$

$$\mathbf{P}^{(j)} = \text{Road-Transformer}(\mathbf{P}^{(j-1)})$$

- 为了更好地捕捉路径中的复杂依赖关系，与掩码语言建模任务类似，我们采用掩码节点建模任务作为一种自监督任务。

$$\mathcal{L}_{\text{mask}} = -\sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{D}} \log \mathrm{P}(v_i | v_i^{\text{mask}})$$

其中，$\mathcal{P}$ 表示所有路径的训练集，$\mathcal{D}$ 是道路路径中被随机掩码的位置，而 v_i^{mask} 是根据被掩码的节点。

II-模态对齐

- 研究者设计了一个在三个不同粒度级别——细粒度、中粒度和粗粒度——上运行的损失函数：
- 细粒度：由于每个图像块可能包含多个节点，节点的编码嵌入和相应图像块的编码嵌入应保持方向一致性。

$$\mathcal{L}_{\text{fine}} = \sum_{p \in \mathcal{P}} \sum_{v_i \in \mathcal{R}(p), \mathrm{L}(v_i) = m_j^{(k)}} \left(1 - \frac{\mathbf{p}_i \cdot \mathbf{h}_j^{(k)}}{\|\mathbf{p}_i\| \|\mathbf{h}_j^{(k)}\|}\right),$$

- **中等粒度**：类似地，为了将道路子路径与图像对齐，构建了以下中等粒度损失函数：

$$\mathcal{L}_{\text{medium}} = \sum_{p \in \mathcal{P}} \sum_{s_i \in \mathcal{R}(p)} \left(1 - \frac{\mathbf{p}_{\text{sep}_i} \cdot \mathbf{h}_{\text{sep}_i}}{\|\mathbf{p}_{\text{sep}_i}\| \|\mathbf{h}_{\text{sep}_i}\|}\right),$$

II-模态对齐

- 粗粒度：由于道路路径和相应的图像路径之间存在唯一的对应关系，因此需要一个更清晰的区分。因此，为粗粒度数据构建了一个对比损失函数。

$$\mathcal{L}_{\text{coarse}} = -\sum_{p\in\mathcal{P}}(\log(\frac{\exp(\text{sim}(\mathbf{p}_{\text{cls}},\mathbf{h}_{\text{cls}})/\sigma)}{\sum_{m^{\text{Neg}}\in\mathcal{B}}\exp(\text{sim}(\mathbf{p}_{\text{cls}},\mathbf{h}_{\text{cls}}^{\text{Neg}})/\sigma)} + \frac{\exp(\text{sim}(\mathbf{p}_{\text{cls}},\mathbf{h}_{\text{cls}})/\sigma)}{\sum_{p^{\text{Neg}}\in\mathcal{B}}\exp(\text{sim}(\mathbf{p}_{\text{cls}}^{\text{Neg}},\mathbf{h}_{\text{cls}})/\sigma)}),$$

- 最后，多粒度损失可以被公式化为

$$\mathcal{L}_{\text{multi}} = \mathcal{L}_{\text{fine}} + \mathcal{L}_{\text{medium}} + \mathcal{L}_{\text{coarse}}$$

II-残差融合

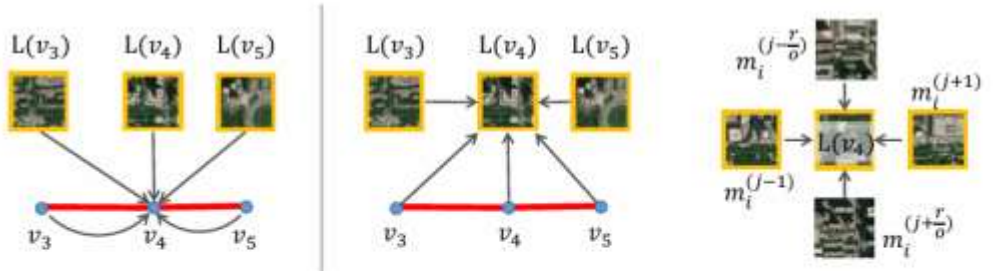
- 将道路初始嵌入与图像编码嵌入进行连接，同时将图像初始嵌入与道路编码嵌入进行连接。由此产生的图像残差嵌入和道路残差嵌入定义为 $U = P^{(0)} \| H$ 和 $Q = P \| H^{(0)}$
- 基于图的融合。我们为每条路径构建一个专门的跨模态有向图

$$\hat{\mathbf{U}} = \text{Relu}\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\text{Relu}\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{U}\mathbf{W}_1\right)\mathbf{W}_2\right)$$

$$\hat{\mathbf{Q}} = \text{Relu}\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\text{Relu}\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{Q}\mathbf{W}_3\right)\mathbf{W}_4\right)$$

$$\mathbf{y} = \text{AvgPooling}(\hat{\mathbf{U}}),$$

$$\mathbf{z} = \text{AvgPooling}(\hat{\mathbf{Q}}),$$



Entities	...	v_3	v_4	v_5	...	$L(v_3)$	...	$m_i^{(j-\frac{r}{o})}$	...	$m_i^{(j-1)}$	$L(v_4)$	$m_i^{(j+1)}$	...	$m_i^{(j+\frac{r}{o})}$	...	$L(v_5)$	...
v_4	0	1	0	1	0	1	0	0	0	0	1	0	0	0	0	1	0
$L(v_4)$	0	1	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0

Figure 4: An example of multi-modal graph construction

II-对比融合损失

- 图像融合嵌入 y 和道路融合嵌入 z 封装了同一路径的多种模态特征，反映了固有的相似性。因此，我们实现了一个四元组损失函数，以确保 y 和 z 之间的差异小于与其他路径融合嵌入的差异

$$\mathcal{L}_{\text{fuse}} = -\sum_{p\in\mathcal{P}} ([||\mathbf{y}-\mathbf{z}||_2^2 - ||\mathbf{y}-\mathbf{z}_{\text{N}}||_2^2 + \beta]_+ + [||\mathbf{y}-\mathbf{z}||_2^2 - ||\mathbf{z}-\mathbf{y}_{\text{N}}||_2^2 + \beta]_+),$$

- 我们的模型的最终训练目标整合了所有先前提出的损失函数，公式化如下：

$$\mathcal{L} = \lambda_{\text{mask}}\mathcal{L}_{\text{mask}} + \lambda_{\text{multi}}\mathcal{L}_{\text{multi}} + \lambda_{\text{fuse}}\mathcal{L}_{\text{fuse}}$$

III Experiment

III Experiment

下游任务：

- 路径旅行时间估计（Path Travel Time Estimation）
- 路径排名得分估计（Path Ranking Score Estimation）

数据集：

我们使用了两个城市的道路网络、全球定位系统（GPS）数据集以及遥感图像数据集，这两个城市分别是丹麦的奥尔堡和中国的西安。道路网络数据来源于OpenStreetMap1，而遥感图像数据集则是从谷歌地球引擎（Google Earth Engine）[15] 获取的

Table 1: Data statistics

	Aalborg	Xi'an
Number of nodes	7,561	7,051
Number of edges	9,605	9,642
AVG edge length (m)	124.78	86.49
Number of path	47,865	200,000
AVG node number per road path	25.77	55.39
AVG path length (m)	3,252.75	4,743.70
Number of traj.	149,246	797,882
Max travel time of traj. (s)	3,549	8,638
Avg travel time of traj. (s)	199	662
Number of images	950	133
AVG number of nodes per image	7.96	53.01
AVG image number per image path	6.28	6.87

III Experiment

实验结果：

Table 2: Overall accuracy on travel time estimation and path ranking

Methods	Aalborg						Xi'an					
	Travel Time Estimation			Path Ranking			Travel Time Estimation			Path Ranking		
	MAE ↓	MARE ↓	MAPE ↓	MAE ↓	τ ↑	ρ ↑	MAE ↓	MARE ↓	MAPE ↓	MAE ↓	τ ↑	ρ ↑
Node2vec [16]	76.228	0.281	54.182	0.203	0.119	0.140	227.129	0.269	30.919	0.218	0.079	0.098
PIM [3]	63.812	0.237	47.054	0.144	0.284	0.343	207.266	0.246	27.716	0.207	0.091	0.102
Lightpath [45]	58.818	0.221	40.219	0.124	0.413	0.483	201.400	0.229	26.429	0.178	0.209	0.252
TrajCL [5]	53.822	0.208	34.239	0.113	0.499	0.577	202.757	0.238	26.506	0.181	0.211	0.256
START [21]	51.176	0.191	34.315	0.117	0.475	0.556	199.843	0.215	25.022	0.179	0.229	0.279
CLIP [37]	72.155	0.261	50.284	0.162	0.179	0.185	219.048	0.256	30.962	0.213	0.087	0.099
USPM [8]	66.714	0.249	51.916	0.148	0.308	0.383	205.594	0.244	26.039	0.209	0.105	0.110
JGRM [29]	51.251	0.193	32.380	0.115	0.512	0.592	201.010	0.228	26.400	0.177	0.228	0.262
Lightpath+image	59.698	0.224	40.920	0.131	0.383	0.405	205.556	0.242	27.058	0.182	0.188	0.231
START+image	51.859	0.188	33.401	0.122	0.437	0.521	200.059	0.211	26.046	0.184	0.183	0.226
MM-Path	**47.756**	**0.172**	**29.808**	**0.106**	**0.558**	**0.643**	**187.452**	**0.193**	**23.644**	**0.165**	**0.257**	**0.294**
Improvement	6.682%	9.947%	12.941%	6.194%	11.823%	11.443%	6.201%	10.236%	5.507%	7.303%	12.227%	5.376%
Improvement*	6.819%	8.511%	7.943%	7.826%	8.984%	8.614%	6.312%	8.531%	9.222%	6.780%	12.719%	12.213%

III Experiment

消融实验:

我们设计了 MM-Path 的八个变体，以验证我们模型中各个组件的必要性：
(1) MM-Path-z：这个变体利用道路融合嵌入 z 作为路径的通用表示。
(2) MM-Path-y：该模型使用图像融合嵌入 y 作为路径的通用表示。
(3) 无对齐：这个版本不包含多粒度损失。
(4) 无融合：这个变体用对两种模态的编码嵌入进行平均池化，来替代基于图的残差融合组件。
(5) 无图卷积网络（GCN）：该模型用交叉注意力机制，替代基于图的跨模态残差融合组件中的图卷积网络。
(6) 无细粒度 (7) 无中粒度和 (8) 无粗粒度：这些变体分别省略了细粒度、中粒度和粗粒度损失。

Table 3: Effect of variants of MM-Path in Aalborg

Methods	Aalborg					
	Travel Time Estimation			Path Ranking		
	MAE	MARE	MAPE	MAE	τ	ρ
MM-Path-z	49.649	0.185	30.193	0.114	0.528	0.622
MM-Path-y	48.529	0.181	32.722	0.118	0.511	0.603
w/o alignment	52.832	0.201	36.251	0.131	0.300	0.379
w/o fusion	51.237	0.192	30.529	0.115	0.476	0.560
w/o GCN	48.651	0.183	33.371	0.111	0.532	0.619
w/o fine	51.641	0.192	33.277	0.129	0.441	0.523
w/o medium	50.932	0.187	34.250	0.114	0.494	0.583
w/o coarse	50.688	0.189	35.341	0.117	0.505	0.596
MM-Path	**47.756**	**0.172**	**29.808**	**0.106**	**0.558**	**0.643**

Table 4: Effect of variants of MM-Path in Xi'an

Methods	Xi'an					
	Travel Time Estimation			Path Ranking		
	MAE	MARE	MAPE	MAE	τ	ρ
MM-Path-z	194.301	0.231	24.455	0.183	0.199	0.241
MM-Path-y	196.331	0.233	24.747	0.196	0.178	0.214
w/o alignment	200.335	0.239	26.459	0.195	0.131	0.167
w/o fusion	200.652	0.239	25.433	0.208	0.113	0.130
w/o GCN	189.659	0.221	24.496	0.173	0.234	0.286
w/o fine	199.214	0.235	26.913	0.177	0.226	0.275
w/o medium	192.514	0.229	24.826	0.175	0.227	0.278
w/o coarse	194.256	0.230	25.757	0.176	0.231	0.278
MM-Path	**187.452**	**0.193**	**23.644**	**0.165**	**0.257**	**0.294**

V Conclusion

V Conclusion

- 本研究提出了一个多模态、多粒度路径表示学习框架，该框架学习适用于各种下游任务的通用路径表示。据我们所知，MM-Path是第一个利用道路网络数据和遥感图像来学习通用路径表示的模型。
- 使用从细粒度到粗粒度的多粒度对齐策略来建模和对齐多模态路径信息。这一策略有效地捕捉了路径的复杂局部细节和更广泛的全局上下文。
- 引入了一个基于图的跨模态残差融合组件。该组件利用跨模态图卷积网络（GCN）来完全整合来自不同模态的信息，同时保持双重模态的一致性。
- 使用两个真实世界的数据集在多种任务上进行了广泛的实验，以展示我们模型的适应性和优越性。

谢 谢！