City Foundation Models for Learning General Purpose Representations from OpenStreetMap

Pasquale Balsebre
Nanyang Technological University
Singapore
pasquale001@e.ntu.edu.sg

Weiming Huang
Nanyang Technological University
Singapore
weiming.huang@nateko.lu.se

Gao Cong
Nanyang Technological University
Singapore
gaocong@ntu.edu.sg

Yi Li
Nanyang Technological University
Singapore
liyi0067@e.ntu.edu.sg

Abstract

Pre-trained Foundation Models (PFMs) have ushered in a paradigm-shift in AI, due to their ability to learn general-purpose representations that can be readily employed in downstream tasks. While PFMs have been successfully adopted in various fields such as NLP and Computer Vision, their capacity in handling geospatial data remains limited. This can be attributed to the intrinsic heterogeneity of such data, which encompasses different types, including points, segments and regions, as well as multiple information modalities. The proliferation of Volunteered Geographic Information initiatives, like OpenStreetMap, unveils a promising opportunity to bridge this gap. In this paper, we present CityFM, a self-supervised framework to train a foundation model within a selected geographical area. CityFM relies solely on open data from OSM, and produces multimodal representations, incorporating spatial, visual, and textual information. We analyse the entity representations generated by our foundation models from a qualitative perspective, and conduct experiments on road, building, and region-level downstream tasks. In all the experiments, CityFM achieves performance superior to, or on par with, application-specific algorithms.

CCS Concepts

• **Information systems** → **Data extraction and integration**; • **Computing methodologies** → **Unsupervised learning**.

Keywords

geospatial data, foundation models, contrastive learning

1 Introduction

The past decade has witnessed a shift from the exclusive provision of geospatial data from national mapping agencies, to freely available, volunteer-based sources. Researchers and practitioners from different disciplines have increasingly recognised the enormous potential of Volunteered Geographic Information (VGI) initiatives, and this trend has been observed in fields such as GIScience, urban planning, cartography and computer science, among others [24]. OpenStreetMap (OSM), one of the most successful VGI projects, aims at building and maintaining a free, editable map of the world. Researchers have harnessed OSM data to train a plethora of algorithms for various tasks, such as traffic analysis [25, 42], land use prediction [23] and recommendation [33]. A typical approach, in the geospatial domain, is to design task-specific algorithms for each downstream application. This presents two major limitations: (1) the models require a large number of labeled samples for training, and (2) the models and representations learned for one task are not necessarily useful for other tasks.

A promising solution to alleviate these limitations lies in adopting Pre-trained Foundation Models (PFMs). A key advantage of PFMs is that the pre-training phase is carried out self-supervisedly, without the need of human annotations; this allows the model to access larger amounts of data and produce effective representations that generalize across tasks. Given the scarcity of labeled geospatial data, and the diverse range of applications it serves, the adoption of PFMs in the geospatial domain presents promising opportunities [29]. However, such adoption is a non-trivial process, due to the fact that geospatial objects are characterized by multi-modal information, including a position in the space, textual annotations, and physical attributes such as shape and size. In addition, geospatial data exhibit inherent heterogeneity, requiring different approaches to handle the diverse entity types. For instance, OpenStreetMap's database stores spatial entities categorized into three types: `Nodes` to represent Points of Interest (POIs); `Ways` are multi-point geometries, comprising polylines and polygons, and are used to represent roads, bridges or large POIs; `Relations` are lists of `Nodes`, `Ways`, or `Relations`, called members, and represent relationships between them. Additionally, each entity can be optionally associated with a set of textual `Tags`, stored as key-value pairs.

Existing studies have primarily focused on a single data type, e.g., entities of type `Way` (polylines), to build a road network for traffic speed prediction [21, 37]. While this approach is intuitive, it fails

Pasquale Balsebre, Weiming Huang, Gao Cong, and Yi Li

to leverage the diverse and multimodal nature of geospatial data types, which convey unique information aspects of the same entity. For example, a large portion of buildings, in OSM, is represented as polygons, but only ~20% is associated with a tag describing its functionality. Previous research works [3, 39, 46] frequently omitted *untagged* spatial entities, considered less informative; yet an important hint on an entity's functionality can be provided by auxiliary information, such as its shape, size and position. Other studies have used supplementary data of different modalities, including human trajectories [8] and street view images [38], to complement OSM data and enhance their framework's performance. However, such data may be expensive to obtain, or available only in specific cities.

In this study, we propose `CityFM`, a framework to train a foundation model within a selected geographical area of interest, such as a city, capable of producing meaningful representations for geospatial entities of different types, which can be easily employed in a wide range of downstream tasks. In adherence to the best practices of data ownership and reproducibility, we design `CityFM` to rely solely on OSM data, which is freely available and accessible globally. `CityFM` is designed as a self-supervised learning architecture, based on mutual information maximization, capable of capturing the textual, visual and spatial characteristics of an entity. Computing effective and meaningful representations for OSM entities is especially challenging, due to the diversity of the available data and the different skills, tools and annotation styles of the contributors, which lead to sparsely and heterogeneously annotated entities. In fact, OSM promotes the so-called *Any tags you like* policy, that lets annotators create their own keys and values for tags of entities they are adding. `CityFM` handles spatial objects of different types and is able to infer missing values, using other aspects of the same object, or entities in the spatial context. Furthermore, previous research works [2, 5, 8, 35, 38, 39] have often overlooked `Relations`, owing to the very distinctive types of information represented. We introduce `Relations` in our self-supervised framework, showing they incorporate information about connectivity and public transports, and can guide the model to recognise important transportation hubs and arterial roads in a city. Finally, we conduct qualitative analyses, and quantitative experiments on a set of road, building and region-level downstream tasks, to demonstrate the utility of the representations produced by our foundation models. In conclusion:

- We propose a new framework, `CityFM`, for general-purpose representation learning in the geospatial domain, using exclusively volunteered multimodal data from OpenStreetMap.
- We design a self-supervised task, based on mutual information maximization, that introduces `Nodes`, `Ways` and `Relations` in the learning framework, to train a geospatial PFM within a selected area of interest. `CityFM`-trained models are multimodal and can integrate an entity's textual, visual and spatial aspects.
- We conduct experiments on road, building and region-level downstream tasks to showcase the effectiveness of the embeddings produced by our foundation models, compared to algorithms tailored for the specific tasks. The code, data, and pre-trained are made available[1].

---

[1]https://github.com/PasqualeTurin/CityFM

2 Impact and Applications

`CityFM` is a framework to pre-train a foundation model, without the need of human supervision, in a region of any shape and size. It stands out as the most comprehensive representation learning framework for geospatial vector data. CityFM's comprehensiveness in encoding various geospatial entities shapes the model to be highly effective. We chose to utilize OpenStreetMap data for pre-training, due to the fact that it is the most complete, open source, geospatial database. Nonetheless, different data sources, with similar geospatial entity types, can be employed in the `CityFM` framework. In our open-source software, released on GitHub[3], we show how `CityFM` can be readily used to produce suitable representations of any JSON-formatted spatial entity, from any source. In our experiments, such representations demonstrated to result in superior performance in an array of downstream tasks, spanning urban sensing, geospatial data management and city digital twins, when compared to existing geospatial foundation models, such as SpaBERT [44], general-purpose embeddings, such as GeoVectors [35], and even methods that were tailored for the specific tasks.

2.1 Predicted Impact

We envision that `CityFM`, and future iterations of pre-trained foundation models, will have a broad impact in geospatial applications.

2.1.1 Urban Applications. First, the quality of the entities' embeddings heavily affects the performance of machine learning models employed in downstream urban task. For instance, in next POI recommendation [9, 27, 43], suitable semantic and positional encodings are pivotal to produce relevant results. In addition, `CityFM` advances the state-of-the-art in road- (polyline) and building- (polygon) based tasks, which can produce a positive impact in many applications, such as population estimation, property price inference and travel time estimation. In Section 5.2, we show how the road segments' representations can be used to infer traffic speed with better accuracy compared to application-specific algorithms.

2.1.2 Geospatial Data Management. `CityFM` foundation models can ease the process of navigating large geospatial databases, with different entity types, providing universal representations. This can support database applications such as region search [45], geographic information retrieval [10, 11, 13], and data integration [4], by reducing the feature extraction cost. For instance, in data integration, algorithms are typically designed to learn an embedding for each object, to represent similarity between them, and be used to compute a matching probability [4]. The PFMs introduced in this paper, generate meaningful and multimodal representations for various object types and positions, that can be readily used to query similar objects. Similarly, Geographic Knowledge Graphs (GeoKGs) [5] can benefit from the representations produced by our models: the high-dimensional embedding of geographic coordinates, can, in fact, lift the quality of the node representation in the KG.

2.1.3 City Digital Twin. Enabling Smart Cities through Digital Twins is a promising path [26]. Being natively multimodal, our models are capable of fusing the wealth of data from different sensors, to provide us with a holistic understanding of our cities. We design a specific task to demonstrate the utility of our visual representations, to predict the functionality of buildings in the city.
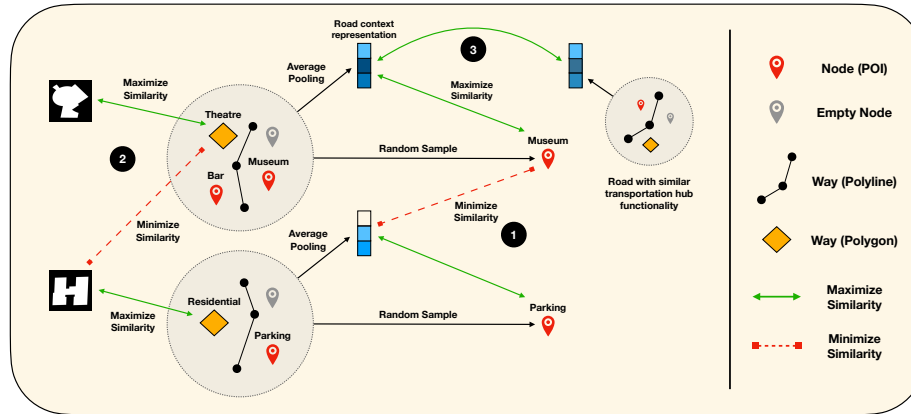
Figure 1: `CityFM`'s self-supervised pre-training framework. The three contrastive objectives are highlighted: (1) Text-based objective; (2) Vision-Language multimodal objective; (3) Road-based objective. Some dashed red lines are omitted for clarity.

The functionality of a building is a crucial feature to predict, for instance, its energy consumption [32]. The embeddings produced by `CityFM` for the different types of entities, can be regarded as a digital replica of the spatial objects in the real world, and can be leveraged by experts for urban planning and land use optimization.

3 Preliminaries

In this section, we provide a detailed description of the data used in this study, and a formal problem definition.

Definition 2.1 (Node): *A* Node *n is a point with a geographic position $n.p = (lat, long)$. It can be associated with a set of tags $n.t = \{t_1, ..., t_m\}$, to represent a Point of Interest. If $n.t = \varnothing$ the node is typically used to construct a more complex geometry, e.g., a polygon, and thus it is not considered as an independent entity.*

Definition 2.2 (Way): *A* Way *w is associated with an ordered list of nodes $w.n = [n_1, ..., n_k]$, that defines its shape as a polyline. If $n_1 = n_k$, w is a closed polygon. Similar to nodes, a* Way *can be optionally associated with a set of tags.* Ways of type polyline are used to represent linear features such as roads, bridges and rivers. Ways of type polygon typically represent larger POIs, such as buildings, universities and airports.

Definition 2.3 (Relation): *A* Relation *r is associated with an ordered list of entities $r.m = [e_1, ..., e_r]$, called members, which can be a combination of nodes, ways and relations. A* Relation *can be associated with a set of tags, and each of its members can be associated with a string, defining its role in the relation.* An example of Relation is a bus loop, where a set of polylines (Ways) defines its path, and the bus stops are POIs (Nodes).

Definition 2.4 (Geospatial Foundational Pre-training): Given a target geographical area (e.g., a city), we aim to pre-train a general model, using OpenStreetMap's geospatial entities in the target area, in a self-supervised fashion. In order to be considered acceptable, such model is expected to generate spatial entity representations that are suitable for use in geospatial downstream tasks.

4 CityFM

`CityFM` is a framework that pre-trains a geospatial foundation model, using OpenStreetMap's entities in a target region of interest. The workflow involves a data preprocessing part, that removes personal information, including phone numbers, URLs and addresses.

4.1 Self-Supervised Learning

In this section, we introduce and provide the underlying motivations for the `CityFM` framework, depicted in Figure 1. As illustrated in Section 3, large-scale geospatial databases, such as OpenStreetMap, encompass diverse entity types and data modalities. While the available information is abundant, textual annotations, such as the functionality of a building or the surface type of a road, are sparsely distributed and vary in their representation, due to the lack of an underlying ontology. This significantly reduces the amount of training data available, and the capability of deep learning algorithms to learn meaningful representations.

The purpose of the self-supervised learning phase, is to pre-train a general model to learn a unified representation of the spatial objects, using the different entity types and information modalities available. We design three different contrastive objectives, using nodes, polylines, polygons and relational information. The first objective is a mutual information-based text-to-text objective, and is used to train a language model for representation learning of the textual part of the entities. The second one is a vision-language contrastive objective, whose purpose is to learn a visual representation of an object's shape, that can indicate its functionality. The third one is a road-based context-to-context objective, that leverages public transportation information, found in entities of type Relation, to identify road segments with similar functionalities.

4.1.1 Text-based Contrastive Objective. A crucial part of a geospatial entity's textual annotation is the information that denotes its functionality, and a common approach in the field is to partition categories into one-hot-encoded classes [23, 39]. In OSM, this is challenging as tags do not follow a pre-defined, structured ontology. For instance, a node could be tagged with the key-value pair "*shop:*

Table 1: A qualitative comparison between BERT and CityFM (following text-based contrastive pre-training).

	BERT	
$Entity_1$	$Entity_2$	Cosine Similarity
$amenity : hospital$	$amenity : hospital$	1.0
	$amenity : cafe$	0.91
	$amenity : restaurant$	0.90
	$amenity : doctors$	0.87
	...	
	$building : residential$	0.78
	$healthcare : clinic$	0.77

	CityFM	
$Entity_1$	$Entity_2$	Cosine Similarity
$amenity : hospital$	$amenity : hospital$	1.0
	$healthcare : clinic$	0.95
	$amenity : doctors$	0.89
	$healthcare : pharmacy$	0.79
	$shop : medical\_supply$	0.74
	...	
	$building : greenhouse$	-0.22
	$tourism : attraction$	-0.23

florist", which is very specific and the key *shop* alone would provide sufficient information. On the other hand, in a polygon tagged as "*amenity: place_of_worship*", the key *amenity* is too generic, and it is used for different classes such as restaurants and hospitals. In addition, certain unique keys or values, devised by individual contributors, are too sparsely used, which hinders the model's ability to learn a meaningful representation. Because of this, we used a pre-trained LM, i.e. BERT [14], to provide an initial representation for the textual part of the entity. Formally, given an entity e, associated with a set of tags $e.t = \{t_1, ..., t_m\}$, where each tag t_i is a key-value pair $k_i : v_i$, its high-dimensional representation $\boldsymbol{h}_e$ is computed using BERT and a 2-layer MLP to map it into a multimodal space:

$$\boldsymbol{h}_e = MLP(Avg(BERT(S_{e.t}))) \qquad (1)$$

Avg is an average-pooling layer to aggregate the BERT representations of the words in the sequence. $S_{e.t}$ is the serialized sequence of *comma-separated* tags, following the standard format for BERT model:

$$S_{e.t} = [CLS]\ t_1,\ t_2,\ ...,\ t_m\ [SEP]$$

The purpose of the first contrastive objective is to push the embeddings of the entities closer in the high-dimensional space, if they are observed in spatial proximity. To achieve this, we group entities located on the same road segment and maximise the similarity between a randomly sampled entity and its context, computed as the average of the entities in the same group. This particular choice is motivated by the expectation that entities situated within the same road segment, exhibit stronger correlations [15]. Given the fact that the entities in a road context are aggregated with an average-pooling layer, we add an *empty node* in each group, to represent emptiness in the context. Such node has a single tag

$n.t = \{context : none\}$. We employ a noise contrastive estimation (NCE) [17] loss function, where negative samples are contexts of entities sampled from other groups, in the same minibatch:

$$\mathcal{L}_{NCE_T} = -\frac{1}{B}\sum_i^B log(\frac{exp(\frac{\boldsymbol{h}_i^\top \boldsymbol{c}_i}{\tau})}{\sum_j^B exp(\frac{\boldsymbol{h}_i^\top \boldsymbol{c}_j}{\tau})}). \qquad (2)$$

In Eq. 2, B is the minibatch size, $\boldsymbol{h}_i$ is the initial representation for entity e_i, computed as in Eq. 1, τ (= 0.5) is a temperature parameter, and $\boldsymbol{c}_i$ is the representation of C_i, the context entity e_i:

$$\boldsymbol{c}_i = Avg(\{\boldsymbol{h}_e : e \in C_i\}) \qquad (3)$$

We train both the language model and the MLP, to minimize $\mathcal{L}_{NCE_T}$.

Table 1 shows a qualitative comparison between BERT and CityFM. Specifically, we report the cosine similarity between the embedding of some pairs of entities, using their categorical tags. We notice that the similarity of BERT representations is heavily affected by the number of words that the entities share, such as *amenity*. In contrast, CityFM, following the text-based contrastive pre-training, is capable of capturing a deeper semantic similarity, based on the spatial co-occurrence of the entities. As expected, entities that are frequently observed in isolation, such as $\{amenity : fuel\}$, demonstrate significantly higher similarity with $\{context : none\}$.

4.1.2 Vision-Language Contrastive Objective. While a large amount of entities, in the OpenStreetMap's database, is represented as a polygon, only a small fraction (∼20%) is associated with tags. Such *untagged* entities have often been considered irrelevant in existing studies [3, 39, 46], and therefore discarded. Nonetheless, valuable insights about a building's functionality can be provided by auxiliary information, such as its shape and size. Motivated by this, we design a *cross-modal* contrastive learning objective, during which our model is trained to produce a representation for a polygon's shape, that is as close as possible to the representation of its functionality, in high-dimensional space. During this pre-training phase, we utilize only polygons associated with tags. Formally, given an entity e, of type Way, associated with an ordered list of nodes $e.n = [n_1, ..., n_k]$, where $n_1 = n_k$, and a set of tags $e.t = \{t_1, t_2, ..., t_m\}$, we compute a high-dimensional representation of its shape,

$$\boldsymbol{s}_e = MLP(ResNet18(Raster(e.n))), \qquad (4)$$

where *Raster* is a *rasterization* function that maps a closed polygon, represented as an ordered list of nodes, to a binary image; *ResNet*18 [18] is a pre-trained vision algorithm that computes a high-dimensional representation of an input image. Since the rasterized polygon covers a fixed portion of the image, independently of its size, we separately compute and embed the surface area of the building as follows:

$$\boldsymbol{a}_e = MLP(\frac{Surface(e.n)}{max_a}). \qquad (5)$$

In Eq. 5, $Surface$ is a function to compute the surface area of a polygon in m^2, and max_a is the maximum area of a polygon, used as a normalization constant. The visual representation of the building, $\boldsymbol{v}_e$, is subsequently computed as the arithmetic mean of its shape
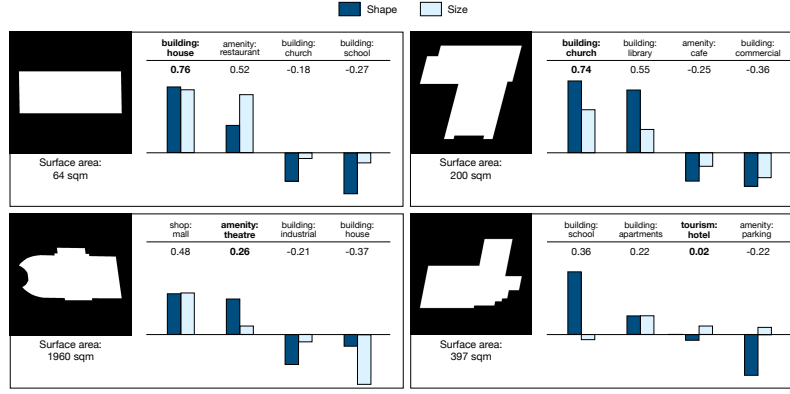
Figure 2: Some examples demonstrating `CityFM`'s capability to associate the visual characteristics of OpenStreetMap's polygons, with their corresponding functionality. We report the similarities of the *shape* and *size* with the textual encodings of the tags.

and size representations, and a 2-layer MLP is used to map it into the multimodal space:

$$v_e = MLP(\frac{s_e + a_e}{2}). \qquad (6)$$

Finally, the following contrastive loss function is used to maximise the similarity, in high-dimensional space, between a building's textual representation h_e, computed as in Eq. 1, and its visual representation v_e, while minimizing the similarity with the visual representations of other polygons in the same minibatch:

$$\mathcal{L}_{NCE_V} = -\frac{1}{P} \sum_{i}^{P} log(\frac{exp(\frac{h_i^\top v_i}{\tau})}{\sum_{j}^{P} exp(\frac{h_i^\top v_j}{\tau})}), \qquad (7)$$

where P is the number of polygons in the minibatch, and τ (= 0.5), is a temperature parameter.

In Figure 2, we showcase `CityFM`'s capability to associate the visual characteristics of OpenStreetMap's polygons[2], with their corresponding functionality. The figure illustrates the similarity of the polygons to some categorical tags in OSM, with the ground truth highlighted in bold font. In addition, we analyze the contribution of the shape and size of the building, by reporting the similarity of their individual representations.

4.1.3 Road-based Contrastive Objective. As defined in Section 3, OSM objects of type `Relation` are characterized by an ordered list of entities, called members, that together describe a more complex object. Relations mostly represent public transportation routes, such as bus loops and train lines. Although public transportation covers the vast majority of urban areas, some roads represent critical links between transportation hubs and key regions of the city. Such roads, often referred to as arterial roads, experience higher traffic volumes and are traversed by a larger number of public means, compared to others. Figure 3 (top) illustrates all the road segments (polyline ways) in the city of Singapore that are members of at least one relation tagged as "*route: bus*", which implies that at least one bus loop traverses the road. In the bottom image of the figure, each road

---

[2]The polygons in Fig. 2 are *tagged* buildings that have been left out during pre-training, for experimental purposes

segment is weighted by the number of bus loops that traverse it. Formally, given an entity e, of type `Way`, associated with an ordered list of nodes $e.n = [n_1, ..., n_k]$, where $n_1 \neq n_k$, its transportation link weight l_e is computed as the number of relations containing e as a member:

$$l_e = |\{r : e \in r\}|, \qquad (8)$$

$$l_e = \frac{l_e}{\max_{j \in W} l_j}. \qquad (9)$$

In Eq. 9, l_e is normalised, and W is the set of all `Way` entities. The second picture in Fig. 3 facilitates the identification of the main transportation hubs in Singapore, and highlights arterial roads that connect different areas of the city. Due to the absence of human mobility data in OSM, we decide to use relations to further refine the representation of road segments, initially based on context alone. Specifically, the embedding h_r of a road segment r, associated with a transportation link weight l_r, is pushed closer, in high-dimensional space, to the embedding of another road with similar transportation link functionality, independently of their distance. In this context, two road segments r_1 and r_2 are considered similar if the difference between their link weights is smaller than a given threshold θ (= 0.05). The set of road segments with similar transportation link functionality to r_e, is therefore defined as:

$$sim(r_e) = \{j : | l_{r_e} - l_{r_j} | < \theta\} \qquad (10)$$

The following loss function is minimized during the road-based contrastive objective,

$$\mathcal{L}_{NCE_R} = -\frac{1}{R} \sum_{i}^{R} \frac{1}{|sim(r_i)|} \sum_{j \in sim(r_i)} log(\frac{exp(\frac{s_i^\top s_j}{\tau})}{\sum_{k}^{N} exp(\frac{s_i^\top s_k}{\tau})}), \quad (11)$$

where R is the number of road segments, s_e is the high-dimensional representation for a road segment e, initialised as in Eq. 3, and N is the number of randomly sampled negatives, with $k \notin sim(r_i)$.
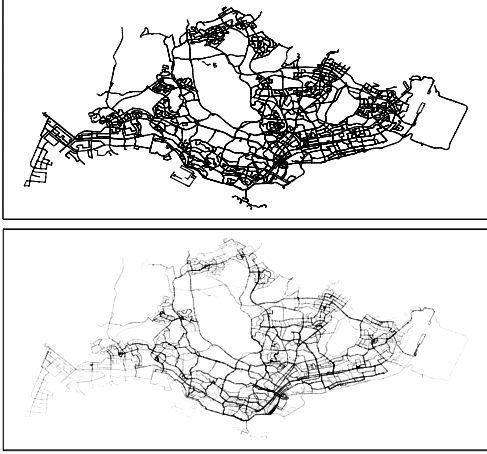
Figure 3: Top: All the road segments traversed by at least one bus loop, in OSM Singapore. Bottom: The same road segments, weighted by number of bus loops traversing them.

4.2 Location Encoding

A geospatial entity is defined by its inherent characteristic of occupying a specific location on the planet, which is typically represented as a 2-dimensional point $p = (lat, long)$. Recent studies have proposed various location encoders, designed to project point-based locations into high-dimensional vectors. In [47], the authors employ a fixed-size grid approach, mapping entities' positions to grid cells and learning high-dimensional representations for each cell. In [12, 22], GeoHash is utilized to translate positions into strings, representing specific cells on the surface of the Earth, and whose size depends on the algorithm's precision, specified by the user. GeoVectors [35] initializes entities' positions with random vectors, and refines them through random walks with distance-dependent transition probabilities, resulting in closer points having similar representations. In Space2Vec [30], the authors encode point-based locations with sinusoidal functions and a multi-layer perceptron, using several frequencies to form a global code-book of 2-dimensional positions. Finally, SpaBERT [44] incorporates the location encoding in the transformer's architecture.

All the aforementioned approaches involve a training phase to learn the parameters of the location encoder. However, considering the vector nature of positional data, where closer positions exhibit higher similarity in the 2-dimensional vector space, we argue that learned parameters are unnecessary. We follow the positional encoding for words in a sequence, introduced in the original Transformer [36] architecture, to define a sinusoidal encoder as follows:

$$f(p_l)^{(i)} := \begin{cases} sin(\omega_k \cdot p_l) & \text{if } i = 2k \\ cos(\omega_k \cdot p_l) & \text{if } i = 2k+1 \end{cases} \quad \forall l = 0, 1 \qquad (12)$$

$$\omega_k = \frac{\lambda}{10000^{\frac{2k}{d}}} \qquad (13)$$

where d (= 128) is the dimension of the encoding space, and λ (= 100) is a rescaling factor that we introduce to facilitate the subsequent learning model's ability to capture the subtle differences that occur between positions expressed in latitude and longitude. In Eq. 12, p_0 is the latitude and p_1 is the longitude, and the two vectors are concatenated, leading to a $2d$-dimensional location encoding. Finally, the wide range of frequencies ensures that the position is represented at different granularity levels.

5 Experiments

The objective of this section is twofold. First, we aim to demonstrate the effectiveness of foundation models pre-trained using `CityFM`, when applied to downstream applications. We compare against baselines that are specifically designed for each task, and trained using downstream data directly. Second, we seek to showcase the potential of utilizing data solely from Volunteered Geographic Information sources, such as OpenStreetMap, which is freely available and accessible globally, and how it can provide valuable information to effectively address geospatial challenges of various type.

5.1 Experimental Settings

`CityFM` models are pre-trained using OSM data, on the three contrastive objectives presented in Sec. 4.1. We minimize the pre-training loss $\mathcal{L}_{pt}$, which is the sum of the task-specific losses:

$$\mathcal{L}_{pt} = \mathcal{L}_{NCE_T} + \mathcal{L}_{NCE_V} + \mathcal{L}_{NCE_R} \qquad (14)$$

The model is trained until convergence with a learning rate of 10^{-4}, a batch size of 256, and a linearly decreasing learning rate scheduler with warm-up. While the framework allows for various text and vision models, we use BERT-base-uncased and ResNet-18. The rasterized images are generated with dimensions of 224x224. Following the pre-training phase, the model's parameters are frozen, and `CityFM` is utilized to generate meaningful representations for the geospatial entities involved in the different downstream applications.

5.2 Traffic Speed Inference

In this *road-based* task, the average speed on each road segment is utilized as the inference objective. The purpose is to evaluate the quality of `CityFM`'s representation of road segments by predicting the average speed at which taxis move on a road link. The data source is Uber Movement, which provides Uber taxi speed data mapped to OSM road segments. Road segments with less than 10 speed measurements are filtered out, obtaining 29,755 data points for NYC, and 6,745 for Seattle.

5.2.1 Baselines. We compare our approach with a set of state-of-the-art road network representation methods:

- **Node2Vec** [16] learns representations of nodes in a graph by increasing the similarity between node pairs within n-hop neighborhoods, obtained through random walks.
- **GCWC** [19] introduces the Graph Convolutional Weight Completion framework, which exploits the road network graph's topology and the correlation among adjacent edges to estimate and fill in the missing weights in the network.

Table 2: Results of Traffic Speed Inference, reporting mean and stddev of 10 independent runs. Speed measure is miles per hour.

Model	NYC				Seattle			
	RMSE ↓	MAE ↓	R^2 ↑	MAPE ↓	RMSE ↓	MAE ↓	R^2 ↑	MAPE ↓
Node2Vec [16]	6.82 (± 0.12)	5.31 (± 0.05)	0.38 (± 0.04)	32.22% (± 0.6%)	7.19 (± 0.08)	6.38 (± 0.07)	0.3715 (± 0.02)	31.24% (± 0.1%)
GCWC [19]	6.74 (± 0.04)	5.2 (± 0.04)	0.4112 (± 0.04)	32.75% (± 1.2%)	7.14 (± 0.06)	5.71 (± 0.09)	0.4437 (± 0.01)	30.95% (± 0.7%)
RFN [21]	6.45 (± 0.09)	4.83 (± 0.02)	0.46 (± 0.02)	30.1% (± 0.01%)	7.22 (± 0.02)	5.69 (± 0.02)	0.51 (± 0.0)	30.7% (± 0.1%)
GeoVectors [35]	5.21 (± 0.07)	3.92 (± 0.06)	0.6423 (± 0.06)	24.03% (± 0.3%)	6.43 (± 0.14)	4.99 (± 0.13)	0.601 (± 0.01)	25.23% (± 1.5%)
IRN2Vec [37]	5.02 (± 0.04)	3.78 (± 0.02)	0.66 (± 0.01)	24.3% (± 0.0%)	6.11 (± 0.26)	4.74 (± 0.19)	0.6298 (± 0.03)	24.2% (± 0.1%)
CityFM	**4.08** (± 0.01)	**3.2** (± 0.01)	**0.7709** (± 0.02)	**19.27%** (± 0.8%)	**4.9** (± 0.05)	**3.79** (± 0.04)	**0.7499** (± 0.02)	**18.18%** (± 0.2%)

Table 3: Ablation study on the Traffic Speed Inference task. We restrict our model's access to only one type of information at a time, to assess their impact. In the last row we exclude the Road-based contrastive objective during pre-training.

Model	NYC				Seattle			
	RMSE ↓	MAE ↓	R^2 ↑	MAPE ↓	RMSE ↓	MAE ↓	R^2 ↑	MAPE ↓
CityFM (PE)	6.99 (± 0.01)	5.13 (± 0.02)	0.2811 (± 0.03)	31.86% (± 0.3%)	8.72 (± 0.04)	6.26 (± 0.05)	0.2184 (± 0.01)	30.55% (± 0.5%)
CityFM (Tags)	6.24 (± 0.19)	5.01 (± 0.11)	0.4537 (± 0.04)	**24.28%** (± 0.5%)	6.34 (± 0.05)	4.81 (± 0.04)	0.4945 (± 0.01)	**23.31%** (± 0.5%)
CityFM (Context)	**5.85** (± 0.01)	**4.76** (± 0.01)	**0.5821** (± 0.02)	24.39% (± 0.8%)	**5.84** (± 0.01)	**4.71** (± 0.01)	**0.609** (± 0.0)	23.97% (± 0.2%)
CityFM (Context w/o Road-based c.o.)	6.06 (± 0.03)	4.99 (± 0.03)	0.5755 (± 0.03)	24.8% (± 0.7%)	5.92 (± 0.02)	4.73 (± 0.03)	0.6065 (± 0.02)	24.4% (± 0.5%)

- Relational Fusion Network, **RFN** [21], is a modified Graph Convolutional Network (GCN), designed for road network settings.
- **IRN2Vec** [37] learns vector representations for intersections of road networks, utilizing random walks to generate sequences of adjacent intersections.
- **GeoVectors** [35] introduces an open corpus of embeddings for OSM entities, which includes tags and location embeddings.
- **CityFM** utilizes road segment tags, positional encoding,and the learned representation based on the entities in the road's context.

5.2.2 Performance Analysis. We report the root mean square error (RMSE), the mean absolute error (MAE), the coefficient of determination (R^2) and the mean absolute percentage error (MAPE), with mean and standard deviation of 10 independent runs, in Table 2. The speed of taxis is measured in miles per hour (mph), and the average speed in NYC is 19.5 mph ($\sigma = 8.19$), while in Seattle is 24.15 mph ($\sigma = 10.18$). CityFM achieves the best results on both the datasets, with a mean absolute error of 3.2 mph in NYC dataset and 3.79 mph in Seattle's. All the baselines, with the exception of GeoVectors and ours, consider the road network as a (un-) directed graph, and use Graph Convolutional Networks. While this approach effectively highlights road connectivity and attribute propagation, it may under-perform in a real world scenario where measurements are limited to only a small subset of road segments, and additional information need to be incorporated.

The ablation study in Table 3, shows that the contextual view of road segments emerges as a strong predictor for inferring the average speed at which taxis travel. The entities surrounding a road provide valuable information on its usage and traffic patterns, and serve as a replacement for missing polyline annotations. The positional encoding (PE), while being the weakest predictor, encodes road segments' absolute positions, allowing `CityFM` to capture spatial relationships between them.

5.3 Building Functionality Classification

As demonstrated in Figure 2, `CityFM` is capable of associating building shapes to their respective functionalities. This is accomplished through the Vision-Language contrastive objective, which leverages the OSM buildings that have tags associated with them. Considering that a significant portion of OSM polygons remains untagged, this represents an opportunity to use `CityFM` to annotate untagged entities, or to assist human annotators in doing so. We chose to evaluate the performance of `CityFM` using Singapore Governmental data[3], which provides detailed land use information at the level of individual buildings. The dataset consists of 64,384 polygons that belong to one of 8 classes. Detailed statistics about the dataset can be found in Table 4. We divided it into training (50%), validation (25%) and test (25%) sets.

---

[3]https://www.ura.gov.sg/maps/?service=MP

Table 4: Statistics of the Building Functionality Classification dataset. Number of instances and percentage of the total.

Functionality	N. of Instances	Percentage
Residential	43,224	67.1%
Industrial	10,431	16.2%
Commercial	5,190	8.1%
Commercial & Residential	1,645	2.5%
Educational	1,427	2.2%
Civic & Community Institution	1,205	1.9%
Sports & Recreation	751	1.2%
Transport	511	0.8%

Table 5: Results of building functionality classification. The baselines are grouped by data type (visual, textual, spatial) they can process. Best overall performance is in bold, best performance using a specific data type is underlined.

Type	Model	macro-F1	weighted-F1	Accuracy
v	ResNet-18 [Frozen]	11.4% (± 0.4%)	51.19% (± 0.8%)	55.33% (± 0.2%)
v	ResNet-18	25.09% (± 0.7%)	64.71% (± 1.1%)	67.14% (± 1.1%)
v	CityFM-Visual	38.26% (± 0.2%)	70.16% (± 0.3%)	65.93% (± 0.2%)
t	BERT [Frozen]	34.02% (± 0.2%)	35.24% (± 0.5%)	43.41% (± 0.6%)
t	BERT	37.56% (± 0.3%)	44.19% (± 0.7%)	51.77% (± 0.7%)
t	CityFM-Textual	49.23% (± 0.6%)	73.42% (± 0.2%)	69.16% (± 0.2%)
s + t	GeoVectors	47.24% (± 1.4%)	64.18% (± 2.1%)	69.49% (± 1.5%)
s + t	SpaBERT	45.06% (± 1.1%)	78.47% (± 2.4%)	75.95% (± 2.7%)
v + t	CLIP	43.03% (± 1.1%)	80.7% (± 1.6%)	74.98% (± 1.2%)
s + v + t	CityFM	**70.1%** (± 1.7%)	**92.75%** (± 1.2%)	**91.93%** (± 1.3%)

Table 6: Category-specific performance.

Functionality	GeoVectors	SpaBERT	CityFM
Residential	76.97%	82.15%	**96.03%**
Industrial	70.57%	74.25%	**94.65%**
Commercial	**88.22%**	66.66%	87.07%
Comm. & Res.	21.08%	18.14%	**46.29%**
Educational	57.98%	46.05%	**73.00%**
Civic & Commun. Inst.	13.82%	12.46%	**24.92%**
Sports & Recr.	31.73%	38.62%	**69.76%**
Transport	21.84%	17.09%	**61.53%**

5.3.1 Baselines. We compare `CityFM` with baselines that are capable of handling textual, visual and/or geospatial information:

- **BERT** [14] is a pre-trained LM that serves as the foundation of `CityFM`'s textual encoding model. We evaluate its performance in two settings: with frozen weights, where only a 2-layers MLP is trained on top of it, and when fine-tuned on the task.
- **ResNet-18** [18] is a pre-trained CNN-based, vision algorithm that serves as the foundation of `CityFM`'s visual encoding model.
- **CLIP** [34] is a contrastive Language-Image pre-trained algorithm. We use it to encode polygons' visual characteristics and the textual information from nearby entities. The model is fine-tuned.
- We use the **GeoVectors** [35] corpus to retrieve the location encoding of the untagged polygon and the GeoVectors-tags embeddings to represent the entities in its spatial proximity.
- We use **SpaBERT** [44] to encode the name attribute of the entities that form the context of the untagged building to be classified.
- **CityFM** is capable of encoding the visual characteristics of a polygon, its position, and the textual features of nearby entities.

5.3.2 Performance Analysis. Table 5 shows the overall best results in bold font, and the best results within a given baseline type underscored. `CityFM`'s textual and visual components, although not fine-tuned on this task, perform better than algorithms that we trained specifically for the it. ResNet-18, when fine-tuned, achieves higher accuracy. GeoVectors [35], SpaBERT [44] and CLIP [34], being able to process multi-modal information, represent strong baselines. `CityFM`, is capable of producing meaningful representations for the textual and visual characteristics of geospatial entities, and can encode positions in the space, achieving the best results.

In Table 6, we illustrate the F1-scores of GeoVectors, SpaBERT and CityFM for each category of the building functionality classification task. We specifically chose to compare to GeoVectors and SpaBERT, as they represent direct competitors in utilizing data from volunteered geospatial information sources, and a pre-training approach to generate general purpose representations. The results reveal a performance gap between the two approaches and `CityFM`.

5.4 Population Density Estimation

In this task, our objective is to estimate the average population density in different regions of NYC and Singapore. The primary goal of this task is to demonstrate that the representations produced by `CityFM` can be used in aggregation to effectively represent regions.

5.4.1 Baselines. The baselines are region embedding methods:

- **Place2Vec** [41] learns entities' representations based on their spatial co-occurrence, inspired by word2vec's algorithm [31].
- **Urban2Vec** was proposed in [38], and learns a low-dimensional representation for each urban region utilizing images and POIs from the region itself.
- We retrieve from the **GeoVectors** [35] corpus the locations and tags encodings for all the spatial entities inside a region, and utilize an average aggregation function.
- **HGI** [20] uses Hierarchical Graph Infomax to learn representations at the POI- and region-levels.
- We use **SpaBERT** [44] pre-trained LM to encode the entities in each region, and we aggregate them by computing the mean of their representations.

Table 7: Estimation of population density.

Model	Singapore			NYC		
	RMSE ↓	MAE ↓	R^2 ↑	RMSE ↓	MAE ↓	R^2 ↑
Place2Vec	10.09 (± 0.3)	7.1 (± 0.21)	0.17 (± 0.05)	10.95 (± 0.35)	8.49 (± 0.24)	0.25 (± 0.02)
Urban2Vec	5.76 (± 0.04)	4.48 (± 0.04)	<u>0.59</u> (± 0.04)	6.31 (± 0.06)	5.94 (± 0.09)	0.66 (± 0.01)
GeoVectors	6.38 (± 0.23)	5.2 (± 0.27)	0.51 (± 0.06)	7.56 (± 0.3)	6.09 (± 0.22)	0.58 (± 0.01)
HGI	<u>5.31</u> (± 0.57)	<u>4.06</u> (± 0.31)	0.57 (± 0.05)	<u>6.15</u> (± 0.23)	**4.18** (± 0.17)	**0.72** (± 0.02)
SpaBERT	6.89 (± 0.68)	4.85 (± 0.43)	0.49 (± 0.05)	8.08 (± 0.19)	5.69 (± 0.16)	0.51 (± 0.03)
CityFM	**4.31** (± 0.22)	**2.75** (± 0.18)	**0.82** (± 0.04)	**5.95** (± 0.27)	<u>4.26</u> (± 0.2)	**0.72** (± 0.02)

- We utilize the representations that **CityFM** produces for the textual annotations of the entities, the visual shapes of the polygons located inside a region, and the position of the region itself.

5.4.2 Performance Analysis. The results are reported in Table 7. Population density is expressed in thousands of people per square kilometer. GeoVectors explicitly encodes textual and spatial information of the entities, producing more meaningful representations. Urban2Vec only implicitly represents spatial proximity, through its contrastive objective. However, it utilizes both textual and visual information, achieving higher performance. SpaBERT underperforms on this particular task, probably due to its inability to capture the proximity among regions. HGI, being a recent algorithm specifically tailored for region-level tasks, performs comparably to `CityFM` on the NYC's dataset.

6 Related Work

6.1 Pre-trained Foundation Models

Despite their success in many domains, there exists very little work exploring the development of PFMs for geospatial artificial intelligence [29]. GeoVectors [35] is an open corpus of OSM entities' representations. It is a framework to efficiently produce embeddings for a representative set of snapshots of OSM, using FastText [7] for the textual annotations and random walks to learn a representation of the entities' locations. Given that the embeddings are task-agnostic, this can be regarded as an initial effort towards general-purpose representations of geospatial entities. More recently, the authors of SpaBERT [44] proposed a framework to train an LLM, initialized with BERT's weights, using geospatial data. SpaBERT is capable of producing representations only for the textual part of OSM nodes. In contrast with `CityFM`, SpaBERT uses entities' names and does not incorporate their tags. Finally, existing works are not designed to handle different entity types.

6.2 Downstream Tasks

6.2.1 Traffic Speed Inference. In this task, the objective is to infer the average speed at which the traffic moves on *unseen* roads. This problem has been studied extensively by researchers. GCWC [19] utilizes only the topology of the road network through GCNs, and

stochastic weight completion to predict edge weights. Relational Fusion Networks (RFNs) [21] were proposed as an alternative to classical GCNs, to specifically model road networks. In previous works the road network has been treated as a graph. In real world scenarios, where measurement are often available for a limited amount of road segments, and road-level features are sparsely distributed, algorithms relying solely on information propagation may not yield satisfactory performance. `CityFM` considers contextual information by utilizing entities located in proximity of the road, and taking into account the road's absolute spatial position, which helps the model learn how roads are used in the city.

6.2.2 Building Functionality Classification. In order to evaluate the quality of our embeddings, we chose to classify the functionality of OSM untagged buildings, given that visual, textual and positional information can be leveraged in this task. This represents a step forward towards automatically annotating geospatial databases. However, this problem has not been studied extensively enough. In [1], the authors use machine learning techniques and a set of features to classify buildings of various types. The approach presents several drawbacks, such as the use of OSM tags and building use codes from external sources, as input features. In [28], researchers used LiDAR features with ML models to to discern between residential and non-residential buildings. Other studies, have harnessed airborne laser scanning data [6] and heat consumption [40]. Existing approaches have utilized information from different sources, which are available only in selected areas, and functionality information of different types, such as OSM tags and land use labels.

6.2.3 Population Density Estimation. Population density estimation in urban regions has crucial consequences in urban planning and resource allocation, and is an important signal of socioeconomic development. Place2Vec [41] is designed to learn POIs categories embeddings, by maximizing the similarity of categories that are often observed in spatial proximity. In contrast, Urban2Vec [38] utilizes multimodal data: the textual tokens of POIs' tags and street view images are mapped to the same embedding space, using a triplet loss function. Recently, HGI [20] was proposed to learn entities' and regions' representations jointly and in a self-supervised fashion. Compared to `CityFM`, existing approaches suffer from two main drawbacks: first, they are tailored specifically for region representation learning and cannot be used in different applications; secondly, they are designed to handle limited multimodal data.

7 Conclusions

This study, presents a novel framework to pre-train foundation models, in a target geographical area of interest. This is a step forward towards a wider adoption of general purpose AI algorithms to address urban challenges. `CityFM` relies exclusively on OSM data, facilitating its adoption and reproducibility. Its multimodal capabilities enable it to access diverse entity types, and learn better representations, using different aspects of each entity, which result in its superior performance.

Acknowledgments

References

[1] Bandam Abhilash, Eedris Busari, Chloi Syranidou, Jochen Linssen, and Detlef Stolten. 2022. Classification of Building Types in Germany: A Data-Driven Modeling Approach. *Data* 7 (04 2022), 45. https://doi.org/10.3390/data7040045

[2] Louai Alarabi, Ahmed Eldawy, Rami Alghamdi, and Mohamed F. Mokbel. 2014. TAREEG: A MapReduce-Based Web Service for Extracting Spatial Data from OpenStreetMap. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data* (Snowbird, Utah, USA) *(SIGMOD '14)*. Association for Computing Machinery, New York, NY, USA, 897–900. https://doi.org/10.1145/2588555.2594528

[3] Ahmed Loai Ali, Falko Schmid, Rami Al-Salman, and Tomi Kauppinen. 2014. Ambiguity and Plausibility: Managing Classification Quality in Volunteered Geographic Information *(SIGSPATIAL '14)*. Association for Computing Machinery, New York, NY, USA, 143–152. https://doi.org/10.1145/2666310.2666392

[4] Pasquale Balsebre, Dezhong Yao, Gao Cong, and Zhen Hai. 2022. Geospatial Entity Resolution. In *Proceedings of the ACM Web Conference 2022* (Virtual Event, Lyon, France) *(WWW '22)*. Association for Computing Machinery, New York, NY, USA, 3061–3070. https://doi.org/10.1145/3485447.3512026

[5] Pasquale Balsebre, Dezhong Yao, Gao Cong, Weiming Huang, and Zhen Hai. 2023. Mining Geospatial Relationships from Text. *Proc. ACM Manag. Data* 1, 1, Article 93 (may 2023), 26 pages. https://doi.org/10.1145/3588947

[6] Mariana Belgiu, Ivan Tomljenovic, Thomas J. Lampoltshammer, Thomas Blaschke, and Bernhard Höfle. 2014. Ontology-Based Classification of Building Types Detected from Airborne Laser Scanning Data. *Remote. Sens.* 6 (2014), 1347–1366.

[7] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomás Mikolov. 2016. Enriching Word Vectors with Subword Information. *CoRR* abs/1607.04606 (2016). arXiv:1607.04606 http://arxiv.org/abs/1607.04606

[8] Yile Chen, Xiucheng Li, Gao Cong, Zhifeng Bao, Cheng Long, Yiding Liu, Arun Kumar Chandran, and Richard Ellison. 2021. Robust Road Network Representation Learning: When Traffic Patterns Meet Traveling Semantics. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management* (Virtual Event, Queensland, Australia) *(CIKM '21)*. Association for Computing Machinery, New York, NY, USA, 211–220. https://doi.org/10.1145/3459637.3482293

[9] Yile Chen, Cheng Long, Gao Cong, and Chenliang Li. 2020. Context-Aware Deep Model for Joint Mobility and Time Prediction. In *Proceedings of the 13th International Conference on Web Search and Data Mining* (Houston, TX, USA) *(WSDM '20)*. Association for Computing Machinery, New York, NY, USA, 106–114. https://doi.org/10.1145/3336191.3371837

[10] Zhida Chen, Lisi Chen, Gao Cong, and Christian S. Jensen. 2021. Location- and Keyword-Based Querying of Geo-Textual Data: A Survey. *The VLDB Journal* 30, 4 (mar 2021), 603–640. https://doi.org/10.1007/s00778-021-00661-w

[11] Gao Cong, Christian S. Jensen, and Dingming Wu. 2009. Efficient Retrieval of the Top-k Most Relevant Spatial Web Objects. *Proc. VLDB Endow.* 2, 1 (aug 2009), 337–348. https://doi.org/10.14778/1687627.1687666

[12] Qiang Cui, Chenrui Zhang, Yafeng Zhang, Jinpeng Wang, and Mingchen Cai. 2021. ST-PIL: Spatial-Temporal Periodic Interest Learning for Next Point-of-Interest Recommendation. *Proceedings of the 30th ACM International Conference on Information & Knowledge Management* (2021).

[13] Ian De Felipe, Vagelis Hristidis, and Naphtali Rishe. 2008. Keyword Search on Spatial Databases. In *2008 IEEE 24th International Conference on Data Engineering*. 656–665. https://doi.org/10.1109/ICDE.2008.4497474

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[15] Giorgos Giannopoulos, Konstantinos Alexis, Nikos Kostagiolas, and Dimitrios Skoutas. 2019. Classifying Points of Interest with Minimum Metadata. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Recommendations, Geosocial Networks and Geoadvertising* (Chicago, Illinois) *(LocalRec '19)*. Association for Computing Machinery, New York, NY, USA, Article 1, 4 pages. https://doi.org/10.1145/3356994.3365504

[16] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable Feature Learning for Networks. *CoRR* abs/1607.00653 (2016). arXiv:1607.00653 http://arxiv.org/abs/1607.00653

[17] Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 9)*, Yee Whye Teh and Mike Titterington (Eds.). PMLR, Chia Laguna Resort, Sardinia, Italy, 297–304. https://proceedings.mlr.press/v9/gutmann10a.html

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. *CoRR* abs/1512.03385 (2015). arXiv:1512.03385 http://arxiv.org/abs/1512.03385

[19] Jilin Hu, Chenjuan Guo, Bin Yang, and Christian S. Jensen. 2019. Stochastic Weight Completion for Road Networks Using Graph Convolutional Networks. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. 1274–1285. https://doi.org/10.1109/ICDE.2019.00116

[20] Weiming Huang, Daokun Zhang, Gengchen Mai, Xu Guo, and Lizhen Cui. 2023. Learning urban region representations with POIs and hierarchical graph infomax. *ISPRS Journal of Photogrammetry and Remote Sensing* 196 (02 2023), 134–145. https://doi.org/10.1016/j.isprsjprs.2022.11.021

[21] Tobias Skovgaard Jepsen, Christian S. Jensen, and Thomas Dyhre Nielsen. 2022. Relational Fusion Networks: Graph Convolutional Networks for Road Networks. *IEEE Transactions on Intelligent Transportation Systems* 23, 1 (2022), 418–429. https://doi.org/10.1109/TITS.2020.3011799

[22] Jiaqi Jin, Zhuojian Xiao, Qiang Qiu, and Jinyun Fang. 2019. A Geohash Based Place2vec Model. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*. 3344–3347. https://doi.org/10.1109/IGARSS.2019.8898375

[23] Noah Johnson, Wayne Treible, and Daniel Crispell. 2022. OpenSentinelMap: A Large-Scale Land Use Dataset using OpenStreetMap and Sentinel-2 Imagery. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 1332–1340. https://doi.org/10.1109/CVPRW56347.2022.00139

[24] Jamal Jokar Arsanjani, Alexander Zipf, Peter Mooney, and Marco Helbich. 2015. *An Introduction to OpenStreetMap in Geographic Information Science: Experiences, Research, and Applications*. 1–15. https://doi.org/10.1007/978-3-319-14280-7_1

[25] Sina Keller, Raoul Gabriel, and Johanna Guth. 2020. Machine Learning Framework for the Estimation of Average Speed in Rural Road Networks with OpenStreetMap Data. *ISPRS Int. J. Geo Inf.* 9 (2020), 638.

[26] Ville V. Lehtola, Mila Koeva, Sander Oude Elberink, Paulo Raposo, Juho-Pekka Virtanen, Faridaddin Vahdatikhaki, and Simone Borsci. 2022. Digital twin of a city: Review of technology serving city needs. *International Journal of Applied Earth Observation and Geoinformation* 114 (2022), 102915. https://doi.org/10.1016/j.jag.2022.102915

[27] Shang Liu, Gao Cong, Kaiyu Feng, Wanli Gu, and Fuzheng Zhang. 2023. Effectiveness Perspectives and a Deep Relevance Model for Spatial Keyword Queries. *Proc. ACM Manag. Data* 1, 1, Article 11 (may 2023), 25 pages. https://doi.org/10.1145/3588691

[28] Zhenyu Lu, Jungho Im, Jinyoung Rhee, and Michael Hodgson. 2014. Building type classification using spatial and landscape attributes derived from LiDAR remote sensing data. *Landscape and Urban Planning* 130 (2014), 134–148. https://doi.org/10.1016/j.landurbplan.2014.07.005

[29] Gengchen Mai, Weiming Huang, Jin Sun, Suhang Song, Deepak Mishra, Ninghao Liu, Song Gao, Tianming Liu, Gao Cong, Yingjie Hu, Chris Cundy, Ziyuan Li, Rui Zhu, and Ni Lao. 2023. On the Opportunities and Challenges of Foundation Models for Geospatial Artificial Intelligence.

[30] Gengchen Mai, Krzysztof Janowicz, Bo Yan, Rui Zhu, Ling Cai, and Ni Lao. 2020. Multi-Scale Representation Learning for Spatial Feature Distributions using Grid Cells. *CoRR* abs/2003.00824 (2020). arXiv:2003.00824 https://arxiv.org/abs/2003.00824

[31] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[32] Razak Olu-Ajayi, Hafiz Alaka, Ismail Sulaimon, Habeeb Balogun, Godoyon Wusu, Wasiu Yusuf, and Muideen Adegoke. 2023. Building energy performance prediction: A reliability analysis and evaluation of feature selection methods. *Expert Systems with Applications* 225 (2023), 120109. https://doi.org/10.1016/j.eswa.2023.120109

[33] Yanjun Qin, Yuchen Fang, Haiyong Luo, Fang Zhao, and Chenxing Wang. 2022. Next Point-of-Interest Recommendation with Auto-Correlation Enhanced Multi-Modal Transformer Network. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 2612–2616. https://doi.org/10.1145/3477495.3531905

[34] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. (2021). arXiv:2103.00020 https://arxiv.org/abs/2103.00020

[35] Nicolas Tempelmeier, Simon Gottschalk, and Elena Demidova. 2021. GeoVectors: A Linked Open Corpus of OpenStreetMap Embeddings on World Scale. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management* (Virtual Event, Queensland, Australia) *(CIKM '21)*. Association for Computing Machinery, New York, NY, USA, 4604–4612. https://doi.org/10.1145/3459637.3482004

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *CoRR* abs/1706.03762 (2017). arXiv:1706.03762 http://arxiv.org/abs/1706.03762

[37] Meng-xiang Wang, Wang-Chien Lee, Tao-yang Fu, and Ge Yu. 2019. Learning Embeddings of Intersections on Road Networks *(SIGSPATIAL '19)*. Association for Computing Machinery, New York, NY, USA, 309–318. https://doi.org/10.1145/3347146.3359075

[38] Zhecheng Wang, Haoyuan Li, and Ram Rajagopal. 2020. Urban2Vec: Incorporating Street View Imagery and POIs for Multi-Modal Urban Neighborhood Embedding. *Proceedings of the AAAI Conference on Artificial Intelligence* 34 (04

2020), 1013–1020. https://doi.org/10.1609/aaai.v34i01.5450

[39] Szymon Woźniak and Piotr Szymański. 2021. Hex2vec: Context-Aware Embedding H3 Hexagons with OpenStreetMap Tags. In *Proceedings of the 4th ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery* (Beijing, China) *(GEOAI '21)*. Association for Computing Machinery, New York, NY, USA, 61–71. https://doi.org/10.1145/3486635.3491076

[40] Michael Wurm, Ariane Droin, Thomas Stark, Christian Geiß, Wolfgang Sulzer, and Hannes Taubenböck. 2021. Deep Learning-Based Generation of Building Stock Data from Remote Sensing for Urban Heat Demand Modeling. *International Journal of Geo-Information* 10 (01 2021). https://doi.org/10.3390/ijgi10010023

[41] Bo Yan, Krzysztof Janowicz, Gengchen Mai, and Song Gao. 2017. From ITDL to Place2Vec: Reasoning About Place Type Similarity and Relatedness by Learning Embeddings From Augmented Spatial Contexts. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (Redondo Beach, CA, USA) *(SIGSPATIAL '17)*. Association for Computing Machinery, New York, NY, USA, Article 35, 10 pages. https://doi.org/10.1145/3139958.3140054

[42] Haitao Yuan, Guoliang Li, Zhifeng Bao, and Ling Feng. 2020. Effective Travel Time Estimation: When Historical Trajectories over Road Networks Matter. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data* (Portland, OR, USA) *(SIGMOD '20)*. Association for Computing Machinery,

New York, NY, USA, 2135–2149. https://doi.org/10.1145/3318464.3389771

[43] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. 2013. Time-Aware Point-of-Interest Recommendation. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Dublin, Ireland) *(SIGIR '13)*. Association for Computing Machinery, New York, NY, USA, 363–372. https://doi.org/10.1145/2484028.2484030

[44] Yao-Yi Chiang Zekun Li, Jina Kim and Muhao Chen. 2022. SpaBERT: A Pretrained Language Model from Geographic Data for Geo-Entity Representation. *EMNLP* (2022).

[45] Liang Zhang, Cheng Long, and Gao Cong. 2023. Region Embedding With Intra and Inter-View Contrastive Learning. *IEEE Transactions on Knowledge and Data Engineering* 35, 9 (2023), 9031–9036. https://doi.org/10.1109/TKDE.2022.3220874

[46] Liming Zhang and Dieter Pfoser. 2019. Using OpenStreetMap point-of-interest data to model urban change—A feasibility study. *PLOS ONE* 14, 2 (02 2019), 1–34. https://doi.org/10.1371/journal.pone.0212606

[47] Ji Zhao, Dan Peng, Chuhan Wu, Huan Chen, Meiyu Yu, Wanji Zheng, Li Ma, Hua Chai, Jieping Ye, and Xiaohu Qie. 2019. Incorporating Semantic Similarity with Geographic Correlation for Query-POI Relevance Learning. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence* (Honolulu, Hawaii, USA). AAAI Press, Article 157, 8 pages. https://doi.org/10.1609/aaai.v33i01.33011270