Inferring building function: A novel geo-aware neural network supporting building-level function classification

Xucai Zhang[a], Xiaoping Liu[b], Kai Chen[b], Fangli Guan[a, c], Miao Luo[d], Haosheng Huang[a*]

[a] Department of Geography, Ghent University, Ghent, 9000, Belgium

[b] Guangdong Key Laboratory for Urbanization and Geo-simulation, School of Geography and Planning, Sun Yat-sen University, Guangzhou, 510275, China

[c] The State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, 430000, China

[d] Department of Architecture & Urban Planning, Ghent University, Ghent, 9000, Belgium

Abstract: Buildings are fundamental components of urban areas and they play a vital role in supporting human activities in daily life. Understanding the actual building functions is essential for many urban applications, such as city management, urban planning, and optimization of transportation systems. Existing studies for inferring building functions are mainly based on a building's own features, and ignore its "geographic context" (e.g., the influences of nearby buildings). This paper introduces a novel geo-aware neural network to infer the functions of individual buildings. To this end, the proposed model integrates information about the built environment and human activity of a target building and its "geographic context". The model further includes a geo-aware position embedding generator and transformer encoders to better capture the complex relationships between buildings. The evaluation results demonstrate that the proposed model outperforms all baselines and achieves a classification accuracy of 90.8%. Meanwhile, the proposed model works well even with a small amount of training dataset and has a good transferability to another urban area. In summary, the proposed model is an effective and reliable approach for inferring the functions of individual buildings and has high potential for city management and sustainable urban planning.

* Corresponding author: Haosheng Huang (haosheng.huang@ugent.be)

Keywords: Building Function, Classification, Geographic Context, Social Sensing, Neural Network

1. Introduction

The world's population has grown gradually over the past decades and will continue to grow in the future. Meanwhile, cities are becoming more densely populated, which means that a reasonable and fair distribution of service facilities is of great importance to sustainable urban development (Xiao et al. 2022, Haberl, Wackernagel and Wrbka 2004). As the main element of urban physical space, buildings play a key role in supporting human activities and fulfilling people's needs in daily life. Understanding the actual functions of individual buildings is essential for city managers, urban planners, and policymakers to make informed decisions about managing existing urban spaces and planning future land-use configuration to improve the quality of life of city residents and create sustainable cities (Srivastava et al. 2020). For example, transportation and land use have been combined to develop an effective planning strategy for sustainable urban development (Liu et al. 2022, Ibraeva et al. 2021). Understanding the building-use configuration of an area is also helpful to analyze its energy consumption owing to the different energy consumption patterns in various building function and time (Xiao et al. 2022). Knowing the spatial distribution of building function also contributes to address the issue of urban heat islands and improve the local climate in urban areas (Rahnama 2021, MacKillop 2012). However, traditional authoritative surveys to identify building function are time-consuming, and cannot be used frequently. Additionally, the authoritative function of a building might sometimes differ from its actual usage by people. Therefore, it is essential to propose a framework to automatically infer the functions of individual buildings.

Inferring building function at a fine level, however, has always been a challenging topic due to the complexity of human-environment interactions and the complexity of extracting useful semantic information. Generally, building function or land use is not invariant, and the actual function is likely to be affected by the demand/supply market. In terms of the complexity of extracting useful semantics, it is hard to extract the function of a building or its service information from remote sensing imagery. While POI data might provide hints of the actual usage of some buildings, they are often jumbled. For instance, many buildings often do not contain any POIs (Deng et al. 2022), while other buildings might have many POIs with different categories. This makes it difficult to directly employ POIs for inferring the functions of individual buildings.

While land use classification of large spatial units (e.g., street blocks, neighborhood area) has been intensively studied in literature, research on building-level function inference is still at an early stage. Several studies employ taxi trajectories and street-view imagery for inferring building function (Niu et al. 2017, Liu et al. 2018, Srivastava et al. 2018, Zhuo et al. 2019, Srivastava et al. 2020), which can only cover buildings along a road. Meanwhile, these studies mainly infer the function of a building based on its own features and ignore its "geographic context" (e.g., the influences of nearby buildings). Additionally, conventional machine learning methods are often employed, but these methods have difficulties in capturing the complex relationships between a building's features.

In recognizing these research gaps, this study proposes a novel geo-aware neural network model for building function identification, which captures the deep-level relationships between a target building and its "geographic context" (i.e., its surrounding buildings). The model makes use of the POI information around the buildings as well as human mobility patterns in these buildings. The proposed model includes a geo-aware position embedding generator and a set of Transformer encoders (Vaswani et al. 2017) to generate an embedding vector that characterizes each building (considering its surrounding context). The embedding vectors are then combined to further infer the functions of individual buildings. Our contributions are three-fold:

1) The proposed model integrates a geo-aware position embedding generator and Transformer encoders to jointly model the built environment and human activity pattern, aiming to capture the complex links between a target building and its "geographic context" in the process of building-level function inference.
2) The evaluation results show that our proposed model significantly outperforms the baselines and has a better classification accuracy than the state-of-the-art studies on building function inference. It achieves a classification accuracy of 90.8%, and a kappa coefficient of 0.87. Meanwhile, the model performs well even when the amount of training dataset is limited, and it also has a good transferability for other urban areas. All these results demonstrate that the proposed model is effective and reliable for inferring the functions of individual buildings.
3) We also show that when inferring the function of a target building, it is important to consider its nearby buildings, as the information regarding these nearby buildings helps to "contextualize" the target building. Meanwhile, considering features of service facility and human mobility of buildings leads to the improvement of the classification accuracies.

2. Related Works

2.1 Land-use classification

Traditionally, modelling the functional information of a city is often based on remote sensing images and on large spatial units, such as traffic analysis zones (TAZs) and big street blocks. Spectral and textural characteristics of remote sensing images are often considered to differentiate land use of different areas (Pacifici, Chini and Emery 2009, Hu and Wang 2013). Recently, deep learning approaches for land-use classification have been developed and shown to achieve high classification accuracy (Huang, Zhao and Song 2018, Tong et al. 2020). However, only considering the spectral and textural characteristics in classification might significantly impact the classification accuracy (Pei et al. 2014). Generally, urban land use is strongly associated with social interaction, making it difficult to differentiate land use according to remote sensing images alone (Tu et al. 2017, Zhang et al. 2021). Therefore, various social sensing datasets and location-based big data (Huang et al. 2021) were introduced into land-use classification, for instance, phone signal data (Pei et al. 2014), taxi GPS trajectory data (Liu et al. 2016), and social media check-in data (Zhang et al. 2020). Meanwhile, multi-source datasets also offer the opportunity for mapping essential urban land use information across a city and even a country (Gong et al. 2020, Zong et al. 2020, Liu and Long 2015). To further improve classification accuracy, deep learning methods have been also applied in recent years to extract deep-level features from remote sensing imagery and social sensing datasets (Srivastava, Vargas-Muñoz and Tuia 2019, Cao et al. 2020, He et al. 2021). However, these studies mainly focus on land use classification in large spatial units, such as big street blocks, neighborhood areas, or TAZs.

2.2 Building function inference

As more datasets become available, fine-grained land-use classification tasks such as inferring functions of individual buildings can be conducted. Ground-based pictures (e.g., street-view images) were introduced for multi-label building functions classification (Srivastava et al. 2018). To obtain more semantic information, deep-level semantics were simultaneously extracted from multi-perspective street-view images to infer building functions (Srivastava et al. 2020) and even from the combination of street-view and remote sensing imagery (Srivastava et al. 2019). However, street-view images are only located along roads, indicating that the buildings away from a road cannot be considered using street-view imagery alone (Zhong et al. 2014). Subsequently, many studies integrated more datasets such as POIs, social media data and taxi trajectory data to infer building functions (Niu et al. 2017, Liu et al. 2018, Zhuo et al. 2019). Additionally, Chen et al. (2017) took advantage of the similarity of human activity pattern in the same building function to infer the function of other buildings using the k-medoids method. However, these studies inferred a target building's function mainly based on its own features, ignoring the influences of other nearby buildings on the target building.

2.3 Neural network embedding for capturing spatial correlations

From a methodological perspective, neural network embedding, originally introduced for natural language processing, provides an opportunity to abridge this gap by allowing more context information to be taken into account when modelling the inner semantics of "words" (Niu and Silva 2021). Such an embedding in neural network, e.g., word2vec, maps a word or term in sentences into a low-dimension vector taking account of the context of the word. The output vector represents the word and its context (Mikolov et al. 2013b). Researchers can then use these embedding vectors to improve the training efficiency and model performance of subsequent regression and classification tasks (Li and Yang 2018). This way of modelling a target's context is interesting and relevant for many geography-related problems, as many spatial objects in space often influence each other, making it important to consider the "geographic context" in many applications.

Yao et al. (2017) first applied the word2vec approach proposed by Mikolov et al. (2013a) to model POI sequences, in which the POI sequences were vectorized to obtain the embedding of different POI categories, taking into account the POI context for further land-use classification. However, this way of modelling POI's context disregards the distance between two adjacent POIs. To accurately model POI's context, each POI pair's distance and check-in pattern in social media should be considered, which can help to address the problem of heterogeneous context (Yan et al. 2017, Wang and Moosavi 2020). While some researchers have deployed these methods to classify the land use in large spatial units (Zhai et al. 2019, Hu et al. 2020), they simply used an average of all categories of the POI embedding vectors in a spatial unit, which fails to portray spatial heterogeneity (Niu and Silva 2021). Additionally, Zhang et al. (2021) employed human activity trajectories to classify land use of street blocks, but ignored the distance-decay influence of two areas. Although previous studies have started to consider "geographic context" for land use classification, there exist two main limitations: 1) Previous studies do not consider the distance between two areas; 2) These studies mainly focus on parcel-level (e.g., TAZs), which cannot be directly used in building-level function classification. In short, applying neural networks to infer building functions while considering the relationships between two buildings is still missing.

2.4 Major studies related to this study

Table 1 shows the major studies related to this study. In summary, although most previous studies involving building-level classification achieve good classification performance, the datasets used cannot cover an entire urban area. For example, taxi trajectory and street-view imagery can only cover buildings along roads. Additionally, current studies classify the function of a building based on its own features, disregarding the influences of nearby buildings (i.e., ignoring the "geographic context" of the target building). This work aims to address these issues, by making use of POIs and human mobility data and introducing a geo-aware transformer encoder model to capture the complex relationships between the target building and its "geographic context" (i.e., its nearby buildings).

Table 1. Major studies related to this study

Study	Spatial units	Number of categories	Data source	Classification methods	Accuracy
Niu et al. (2017)	Building level	10	POI, Tencent location data, Taxi trajectory	Density-based method	72%
Liu et al. (2018)	Building level	7	POI, Tencent location data, Taxi trajectory	Probabilistic model	85%
Zhuo et al. (2019)	Building level	5	Tencent location data, Taxi trajectory	Iterative clustering method	85%
Srivastava et al. (2020)	Building level	16	Street view imagery	CNN-based deep learning method	63%
Deng et al. (2022)	Building level	5	POI, remote sensing imagery, street view, building physical properties, land use map	XGBoost	85%
Yao et al. (2017)	TAZ level	14	POI	Word2vec	87%
Zhai et al. (2019)	Neighborhood area	7	POI	Place2vec	70%
Zhang et al. (2021)	Street block	5	POI, mobile phone positioning data	Word2vec and random forest	77%
Our	**Building level**	**12**	**POI, Tencent mobility location data**	**Transformer-based deep learning**	**91%**

3. Study Area and Dataset

3.1 Study area

Figure 1 shows the study area located in the Nanshan district of Shenzhen city in Guangdong province, China. Since performing reform and opening door policy, Shenzhen became the first Special Economic Zone and is currently one of the most successful cities in China. Our study area Nanshan district, with a total area of 187.53 km^2, is the most prosperous and developed district in Shenzhen. We selected the Nanshan district as the study area, since it is filled with a variety of facilities and buildings.

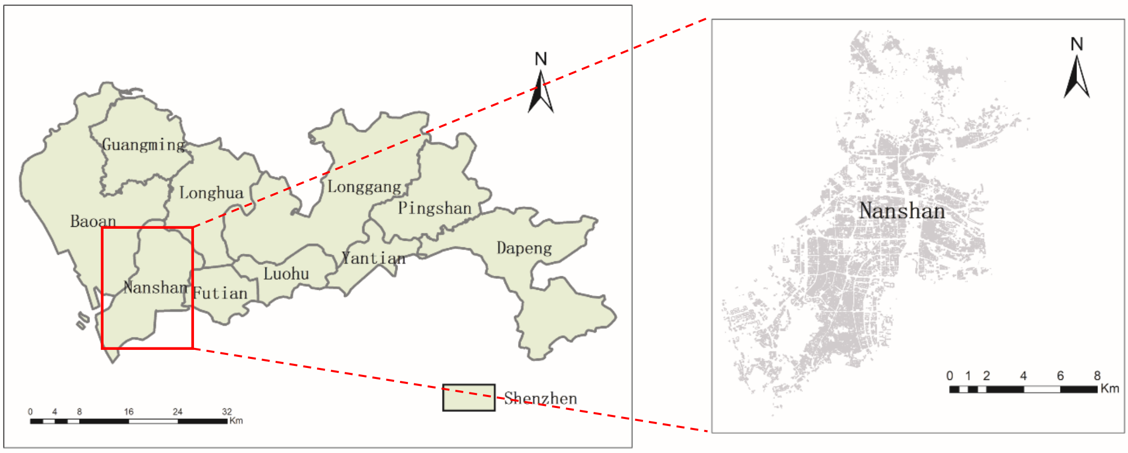


Figure 1. The study area: Nanshan District, Shenzhen, China.

3.2 Datasets

3.2.1 Building footprint data

The building footprint dataset used in this study was collected from Gaode Map (http://gaode.com), which is a popular map service in China. The dataset contains not only the building footprint but also the number of floors of each building. To assign functions to each building, we refined the land use categories based on the Chinese Standard of Land Use Classification. We then assigned the refined categories to all buildings by manually checking the corresponding high-resolution images and buildings' name in the year of 2019, following the procedure defined in Liu et al. (2018). Table 2 shows the refined land use categories.

Table 2. Building category schemes.

Categories	Descriptions
001 Urban village (UV)	Buildings of rural residential housing surrounded by urban blocks
002 Urban residential (UR)	Modern buildings for residential housing
003 Business office (BO)	Buildings of financial, internet, insurance offices

004 Big Catering (CA)	Buildings for catering
005 Shopping centre (SC)	Buildings for big shopping centres
006 Hotel (HT)	Buildings for hotels
007 Recreation & Tourism (RT)	Buildings for entertainment and tourism
008 Company & Factories (CF)	Buildings for small companies and factories
009 Industrial Park (IP)	Buildings for big industrial park
010 Administrative (AM)	Buildings for government and public service agencies
011 Education (EDU)	Buildings for schools
012 Medical (MD)	Buildings for hospitals and healthcare

3.2.2 Point of interest and Tencent location datasets

Point of interest (POI) datasets often contain information of the services offered by buildings. Hence, this study employed the POI dataset, which is collected from Gaode Map in the year of 2018, as one of the main features to portray buildings' function. Gaode Map divided all POIs into 20 categories: auto service, auto dealers, auto repair, motorcycle service, catering, shopping, life service, sport & recreation, medical service, hotel, tourist attraction, residential, administrative organization, education science & research, transportation service, financial & insurance, company, road-related facilities, road & place signs, and small public facilities. Due to the existence of huge redundant information and inappropriate category for inferring building function in the POI dataset, we combine the four categories of auto service, auto dealers, auto repair and motorcycle service as automotive service. We remove the categories of transportation service, tourist attraction, road-related facilities, road & place signs, and small public facilities, as they are normally not linked to individual buildings. Therefore, we finally reclassified the dataset using the following 12 POI categories: catering, company, shopping, financial & insurance, education & research, automotive service, residential, life service, sport & recreation, medical service, administrative organization, and hotel.

Given that buildings with different functions present different human activity patterns, we thus introduced Tencent location dataset, collected from the big data platform of Tencent (https://heat.qq.com), to represent the temporal characteristics of human activity in buildings. The dataset recorded all location requests from users of a variety of Tencent's location based services, including social media, gaming, travel, online shopping, communications and payment tools. Given the ubiquity of Tencent users in China, the Tencent location dataset has a better user representativeness than other alike data (e.g., Twitter data, Weibo data, taxi trace data, and bike-sharing data), and therefore can better reflect the real human activity in a city. The Tencent location dataset used in this study was collected in 2015 with a spatial resolution of 25m × 25m at a 1-hour interval, divided into two types of days: weekdays and weekends. Since there was no big construction change in the Nanshan district from 2015 to 2019, using the Tencent dataset in 2015 to represent human behavior pattern is acceptable.

4. Methodology

4.1 Problem definition

This study aims to infer a building's function based on its services provided and human activity patterns, as well as those of its nearby buildings. Supposing every building *b* can be described as two vectors from the perspectives of service facilities and human activity pattern, respectively: $X_b^S = [x_b^1, \dots, x_b^i, \dots, x_b^s, x_b^f, x_b^m, x_b^b, x_b^p]$ and $X_b^H = [x'^{1}_{b}, \dots, x'^{j}_{b}, \dots, x'^{t}_{b}]$, where *s* is the number of POI categories (in this work $s = 12$; see Section 3.3.2) and x_b^i represents how many POIs of the *i*th POI category are located within the building b, wherein $i \in [1, s]$. Additionally, x_b^f denotes its number of floors; x_b^m refers to the Euclidean distance between building b and its nearest metro station; x_b^b represents the Euclidean distance of that building to its nearest bus station; x_b^p indicates the distance of that building to its nearest park. x'^{j}_{b} represents how many humans were active in this building at j^{th} time, wherein $j \in [1, t]$ and $t = 48$ in this work. This study aims to use $(X_b^S, X_{b_1}^S \dots, X_{b_n}^S)$ and $(X_b^H, X_{b_1}^H, \dots, X_{b_n}^H)$ for inferring the function of the building b, where $b_1 \dots, b_n$ are its n-nearest buildings (using Euclidean distances between the building centers) and serve as its "context". Hence, the problem of building function inference can be defined as:

$$\widehat{y_b} = f\left(X_b^S, X_{b_1}^S \dots, X_{b_n}^S, X_b^H, X_{b_1}^H, \dots, X_{b_n}^H\right) \tag{1}$$

Where the output $\widehat{y_b}$ is a building function category in Table 2, e.g., "006 Hotel (HT)"; the hyperparameter n denotes the number of nearby buildings to be additionally considered.

To solve such a problem, we employ neural network approaches. From a high-level perspective, neural network is a model that consists of layers of neurons that are interconnected to allow information to flow among layers while being processed. Because of the non-linearity introduced by the activation functions, neural networks can capture the complex relationships in the data (e.g., the relationships between buildings and the environments), and may handle complex problems (e.g., inferring functions of individual buildings, as in this work).

4.2 Framework of the proposed model

Figure 2 demonstrates the framework of our model, which mainly consists of two parts: the left one is proposed to characterize the service facilities offered by the target building (i.e., building b) and its nearby buildings (i.e., the "context" of building b), and the right one aims to portray the human activity characteristics of these buildings. The architecture of these two parts is the same. Both include a fully connected feature embedding layer, a geo-aware position embedding generator (Geo-PEG), multiple transformer encoders, and a gated recurrent unit (GRU) based fusion. These two parts help to capture the relationships between the target building b and its nearby buildings (i.e., its “context”), which are then used to infer the function category of the target building b.

Take the left part (i.e., for the service facilities) as an example. A fully connected layer is firstly applied to transform the sparse input feature vectors of the $n + 1$ buildings (i.e., $X_b^S, X_{b_1}^S \dots, X_{b_n}^S$) to dense vectors, with the aims to improve computational efficiency. The results are then fed into Geo-PEG to assign positional and geo-contextual information to every building, taking into account the distance-decay influence and geographical distribution difference. Such Geo-PEG solves the problem of constant contextual distance. With this, an embedding vector for each building is generated.

These embedding vectors are then fed into a set of transformer encoders, each of which is a self-attention mechanism translator model developed by Vaswani et al. (2017). The transformer encoders are used to capture the correlations among these buildings based on distance and semantic. The resulting embedding vector for each building can then better capture the links between the target building and the others, and therefore can be considered as a “contextualized” embedding vector for a building.

At the next step, these refined embedding vectors are combined via a GRU and two fully connected layers to create a fused vector for the target building b. This fused vector captures the key information between the target building and its nearby buildings (i.e., its “geographic context”). The fused vectors of the left (i.e., in terms of service facilities) and right (i.e., in terms of human activity patterns) parts are then concatenated and inputted into a fully connected layer to infer the function category of the target building b.

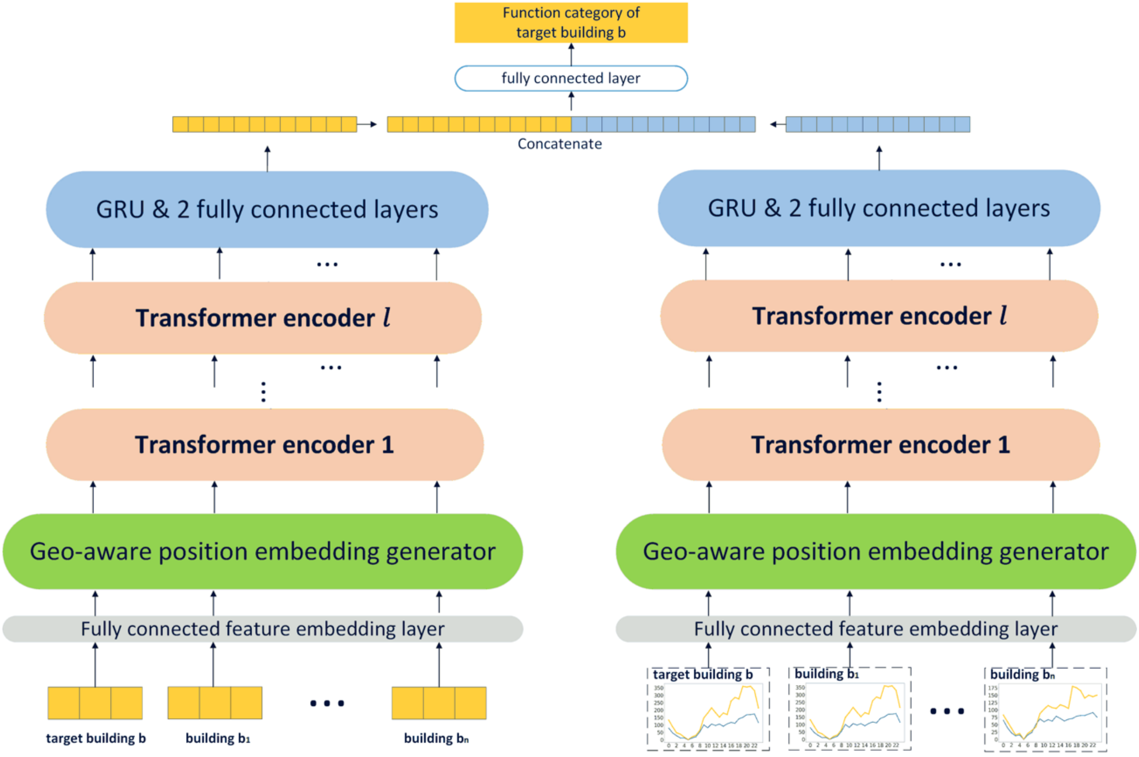

Function category of target building b
fully connected layer
Concatenate
GRU & 2 fully connected layers
GRU & 2 fully connected layers
Transformer encoder l
Transformer encoder l
Transformer encoder 1
Transformer encoder 1
Geo-aware position embedding generator
Geo-aware position embedding generator
Fully connected feature embedding layer
Fully connected feature embedding layer
target building b
building b₁
building bₙ
target building b
building b₁
building bₙ

Figure 2. Framework of the proposed model.

4.3 Fully connected feature layer

Fully connect neural network (FCN) is the most popular and classic neural network in deep learning. To improve computational efficiency, this study applies FCN to transfer the sparse input feature vector of each individual building to a dense vector, which can be described as:

$$X_k^{FCN} = \emptyset(W_{FCN} \times X_k + b_{FCN}) \tag{2}$$

Where W_{FCN} denotes to the learnable weights in FCN; b_{FCN} represents the learnable biases; X_k is the input feature vector (either the service facilities vector X_k^S or the human activity pattern vector X_k^H) of building $k \in \{b, b_1 \dots, b_n\}$, which is one of the $n+1$ buildings. $\emptyset$ refers to the activation function, we use *relu* activation function in this study. In short, FCN is applied to each building's *own* input feature vector to create a refined feature vector X_k^{FCN}. However, the learnable parameters W_{FCN} and b_{FCN} are shared by all buildings. In this work, we define the dimensions (i.e., the dimension of X_k^{FCN}) of the refined vectors of service facilities and human activities as hyperparameters.

4.4 Geo-aware position embedding generator (Geo-PEG)

Taking the refined input feature vectors of the target building and its nearby buildings as inputs, the Geo-PEG component (Figure 3) aims to embed positional and geo-contextual information among the buildings into these vectors. This is needed, as the transformer encoders used afterwards need such positional information to account for the "ordering" of the "words" (i.e., individual buildings in this work) in the input sequence.

Conventionally, the position index of each word is assigned according to their appearance in the sequence, for instance, the first position (i.e., the target building b) is assigned as 0, the second position (i.e., b_1, the nearest building to the target building) is assigned as 1, and the (n+1)th (i.e., b_n, the n-nearest building) is assigned as n. This conventional way (typically used in natural language processing), which is one dimensional and assumes uniform spacing between "words", is not sufficient to capture the spatial configurations between buildings in 2D spaces.

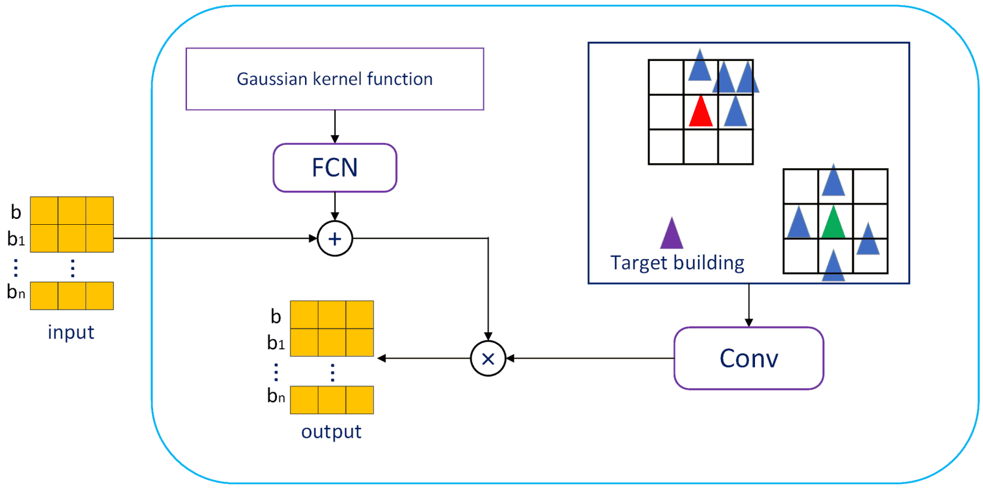


Figure 3. Architecture of geo-aware position embedding generator.

To address this issue, a new approach considering the spatial distances between buildings and spatial distribution around individual buildings is proposed (Figure 3). Firstly, considering distance-decay influence, we apply Gaussian kernel function to assign initial position indexes for all buildings, based on their Euclidean distances to the target building b.

$$g_k = \exp\left(-\frac{d_{k,b}^2}{2 \times \sigma^2}\right) \tag{3}$$

Where $k \in \{b_1 \dots, b_n\}$ denotes a building surrounding the target building b. $d_{k,b}$ is the Euclidean distance between buildings k and b. σ is a bandwidth and a trainable parameter to determine how far away a building still needs to be considered.

Each of these initial position indexes, i.e., g_k, is then added to the input feature of its corresponding building k after forward propagation using a fully connected network layer (FCN). The result of this step is a feature vector enriched with "spatial distance" index for each building.

Secondly, we add more information regarding the spatial distribution around each individual building. Suppose there are two buildings (red and green triangles in Figure 3) that have the same Euclidean distance to the target building b, while these two buildings have different spatial distributions around them. For instance, the red building has its surrounding buildings only on one side, and the green building is surrounded uniformly by other buildings. In this case, it is problematic to assign a same position index for these two buildings from the Gaussian kernel function. Hence, we apply a convolutional layer to portray the spatial distribution around each building and assign an additional weight for every individual building. The convolutional layer can be described as:

$$c_k = f(W_{CONV} * D_k + b_{CONV}) \tag{4}$$

Where * denotes to convolutional operation, W_{CONV} are learnable parameters, and b_{CONV} represents the learnable biases. f indicates the activation function, using sigmoid activation function in this study. The input D_k represents the number of buildings in each of the 3×3 cells around building $k \in \{b_1 \dots, b_n\}$. This 3×3 kernel size, which is commonly applied in many studies, is chosen considering the following two aspects: 1) The nearest buildings provide the most relevant "geographic context" to the target building; 2) The number of trainable parameters grows quadratically with the kernel size, which makes big kernels not cost-efficient.

c_k is the output weight considering spatial distribution around building k. Note that this convolutional operation is applied to each of the n buildings, sharing the same learnable parameters W_{CONV} and b_{CONV}. The spatial distribution weight of each building c_k is integrated into its input feature vector (enriched with "spatial ordering" index g_k; from the previous step) to create a final input feature vector for each building. Each final input feature vector embeds the information regarding the "spatial ordering" and surrounding spatial distribution of its owning building.

In summary, the Geo-PEG component can be described as:

$$X_B' = (X_B + FCN(G)) \times C \tag{5}$$

Where $X_B = [X_b^{FCN}, X_{b_1}^{FCN} \dots, X_{b_n}^{FCN}]$ are refined feature vectors (results of equation (2)) of the target building b and its n-nearest buildings $b_1 \dots, b_n$. $G = [1, g_{b_1} \dots, g_{b_n}]$ are the "spatial distance" indexes (results of equation (3)), and $C = [1, c_{b_1} \dots, c_{b_n}]$ represent the "spatial distribution" weights of each building (results of equation (4)). Note that the first element in both G and C is 1, as they both refer to the targe building itself.

The output of the Geo-PEG component X_B' is a set of embedding vectors, each for the target building b and its n-nearest buildings $b_1 \dots, b_n$. These embedding vectors consider the "spatial ordering" of their corresponding buildings. They will serve as input for a follow-up transformer encoder component.

4.5 Transformer encoder

This module is applied to model the complex relationships between a target building and its nearby buildings. It outputs an embedding vector for each building, which captures the building's characteristics and its links to other buildings. From a high-level perspective, the resulting vector of a building can be considered as a "data point" in an embedding space. Such an embedding space can be used to compute the "distance" between any two buildings.

Taking the embedding vectors X_B' of the target building b and its n-nearest buildings from the previous step as input, the transformer encoder component outputs another embedding vectors X_B^E that capture the correlations between these buildings. The transformer encoder component is a stack of encoder layers. The number of encoder layers l is a hyperparameter. The first encoder layer takes X_B' as input, and its output is then fed to the next encoder layer.

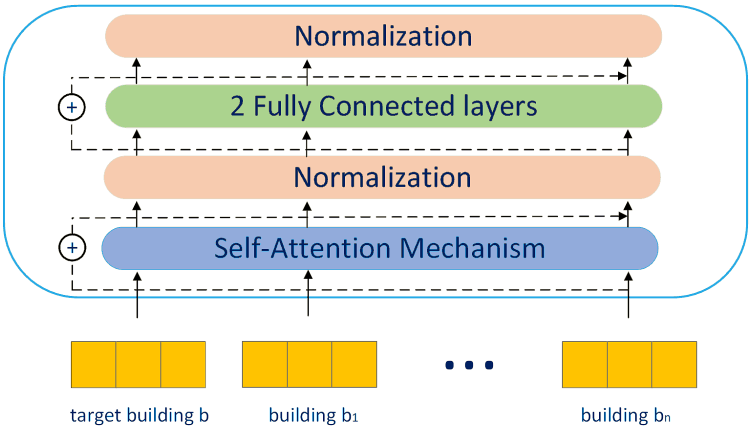


Figure 4. The architecture of a transformer encoder.

Figure 4 shows the architecture of a transformer encoder layer. The transformer encoder used in this study is a self-attention mechanism translator model developed by Vaswani et al. (2017), including a layer for self-attention operation, two normalization layers, and two fully connected layers. Additionally, the transformer encoder applied a shortcut connection trick, namely deep residual learning framework. To capture more details and deeper relationships between any two buildings based on distance and semantic, we use a multi-head self-attention mechanism. The detail can be described as followed (Vaswani et al. 2017):

$$Q = W_Q \times X_B' \tag{6}$$

$$K = W_K \times X_B' \tag{7}$$

$$V = W_V \times X_B' \tag{8}$$

$$head_m = softmax(\frac{Q_{head_m} K_{head_m}}{\sqrt{d}}) V_{head_m} \tag{9}$$

$$X_B^A = MultiHeadAttention(Q, K, V) = Concat(head_1, \ldots, head_m)\ W_M \tag{10}$$

Where Q, K and V are the query, key, and value for operating attention score respectively, and they are obtained by multiplying different learnable weights with output of the previous Geo-PEG generator. Q_{head_m} , K_{head_m} and V_{head_m} are gained from dividing Q, K and V into *m* parts for m-head component. Softmax is used as the activation function to map the vector into [0,1] in which the sum of every dimension is 1. $MultiHeadAttention(Q, K, V)$ is a function to concatenate all heads and subsequently multiply heads by a learnable weight W_M. X_B^A indicates the output of multi-head attention, capturing the relationships between buildings.

Summarily, the detail of the transformer encoder component can be described as:

$$X_B^{E^1} = Normalize(2FCN\ (Nrmalize\left(X_B' + X_{\mathrm{B}}^{A^1}\right))) \tag{11}$$

$$X_B^{E^i} = Normalize(2FCN\ (Nrmalize\left(X_B^{E^{i-1}} + X_{\mathrm{B}}^{A^i}\right))) \tag{12}$$

Where X_{B}^A is the output of the multi-head self-attention (equation (10)), and $X_B^{E^i}$ is the output of layer $i \in [2, l]$. The first encoder layer (equation (11)) takes X_B' as input, and its output is then passed into the next encoder layer. All other encoder layers take the output of their previous layer (equation (12)).

The output of the last encoder layer $X_B^{E^l}$ is the refined embedding vectors for the target building b and its *n*-nearest buildings. Each refined embedding vector captures the relationships between its corresponding building and all other ones.

4.6 Gated recurrent unit (GRU)

The refined embedding vectors are then fed into a GRU layer and two fully connected layers to create a fused vector for the target building b. GRU is a type of recurrent neural network capable of capturing important information of a long sequence of "words" (in this work, each refined embedding vector can be considered a "word"). The fused vector, outputted by GRU, captures the key information between the target building and its nearby buildings (i.e., its "context"). This process can be described as:

$$X_b'' = 2FCN\left(GRU\left(X_B^{E^l}\right)\right) \tag{13}$$

The GRU layer contains $n + 1$ GUR units, corresponding to the target building b and its *n*-nearest buildings. The first GRU unit takes the refined embedding vector of the farthest building as input, and the last GRU unit takes the refined embedding vector of the target building b as input. All other buildings, ordered from the farthest to the closest to the target building, correspond to the other GRU units in between.

A single GRU unit (Figure 5) takes the current input data X_j and state h_{j-1} (which contains the useful information of the previous $j - 1$ GRU units), and outputs the new state h_j. It has two gates: an update gate that determines how much of the previous information needs to be passed to the future (i.e., h_j); a reset gate deciding how much of the previous information to forget. The calculation formula of each GRU unit is shown below:

$$r_j = \sigma(W_r \times [h_{j-1}, X_j] + b_r) \tag{14}$$

$$u_j = \sigma(W_u \times [h_{j-1}, X_j] + b_u) \tag{15}$$

$$c_j = tanh(W_c \times [r_j \circ h_{j-1}, X_j] + b_c) \quad (16)$$

$$h_j = (1 - u_j) \circ h_{j-1} + u_j \circ c_j \quad (17)$$

$$\widehat{U}_{j+1} = \sigma(W_o \times h_n) \quad (18)$$

Where X_j denotes the current input data (e.g., the refined embedding vector of building $j \in [0, n]$), W represents the learnable parameters, σ and $tanh$ refer to sigmoid and hyperbolic tangent activation functions which add nonlinearities to the model, operator $\circ$ denotes Hadamard product (i.e., element-wise multiplication). r_j and u_j denote the reset gate and update gate respectively, which control how much previous information h_{j-1} (gained from previous time steps) and current information gained from X_j will be passed to the new state h_j. c_j is a candidate state.

Finally, the output $\widehat{U}_{n+1}$ is fed into the two fully connected layers to create a fused vector X_b'' for the target building b.

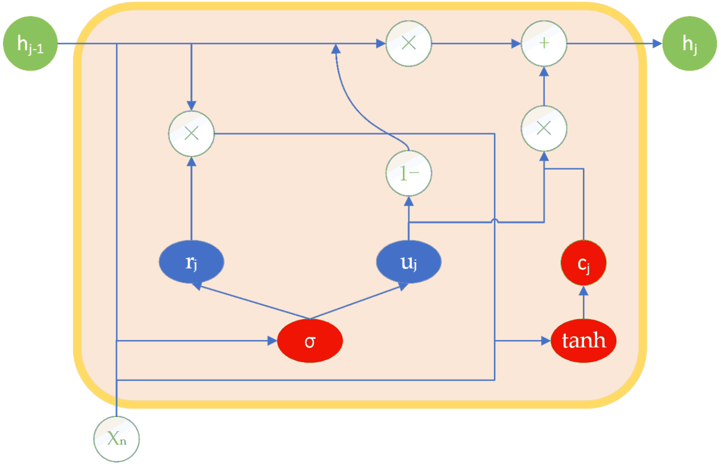


Figure 5. The workflow of a GRU unit

4.7 Final component and loss function

Finally, the fused vector of the left part (i.e., in terms of service facilities, $X_b^{S''}$) and the right (i.e., in terms of human activity patterns $X_b^{H''}$), i.e., the output from the previous part (equation (13)), are concatenated into a single vector. The concatenated vector is then fed into a fully connected layer to infer the function category of the target building b. This can be described as:

$$P(category) = softmax(FCN(concat(X_b^{S''}, X_b^{H''}))) \tag{19}$$

The target building b is then assigned the function category (i.e., a category from the list in Table 2; e.g., "Hotel") that has the highest probability value. The loss function used in this study is multi-categories cross entropy, which can be described as:

$$Loss = -\sum_{i=1}^{K} y_i \log(p_i) \tag{20}$$

Where K is the number of building function categories. y denotes to the ground truth. If category i is the truth label, then y_i=1. Otherwise y_i=0. p_i represents i^{th} category's probability inferred by the model.

4.8 Example: the workflow of inferring the function of an example building

Figure 6 shows the workflow of inferring the function of an example building. Suppose the red building is the target building whose function is to be inferred, the blue ones are its 2 nearest buildings being considered. Here we take the service facilities part as an example. The input vectors like [2, 3, …, 20] mean how many POIs are located in the building, its number of floors, and its distances to the nearest bus, metro and park. Thus, for these three buildings, a matrix shaped as (3, 16) can thus be constructed. Subsequently, the matrix passes through the fully connected feature layer (Section 4.3), and each of its three vectors is transformed into a dense vector, with its dimension being changed to a certain value (Here, we select 25 for this sample. Section 5.2 describes how the value is selected). After the Geo-PEG (Section 4.4), the information of distance and distribution pattern is added into the matrix. Processing the matrix, the transformer encoder stack (Section 4.5) then outputs an embedding vector for each building, which captures the building's characteristics and its links to other buildings (i.e., its "geographic context”). The shape of the output matrix of the transformer stack is also (3, 25). Subsequently, GRU module (Section 4.6) is used to augment the geographic context again by aggregating the matrix to a vector (1, 25). This vector of service facilities, and the other vector (1, 36) of human activity pattern, are then concatenated into a joint vector in the form of (1, 61), and fed into a fully connected layer with “softmax” (Section 4.7) to infer the building function. The output is a list of probability values of the 12 categories (Table 2) to which the target building belongs. The category with the highest probability value is the inferred function (e.g., “001 Urban village (UV)”) of the target building.

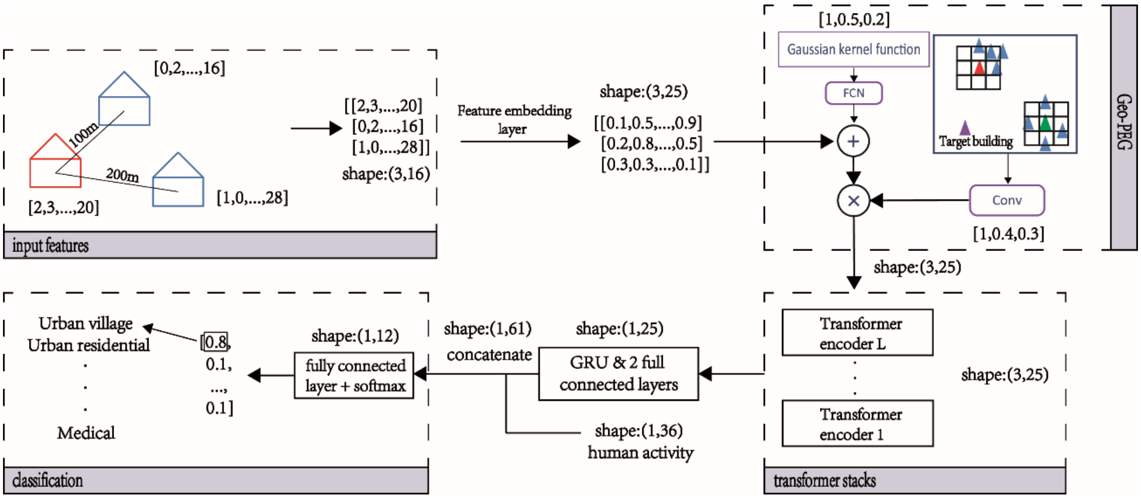


Figure 6. The workflow of inferring the function of an example building (marked in red)

5. Evaluation and Results

5.1 Evaluation setup

In this section, we evaluate the proposed model with several other baselines, using the dataset of the Nanshan district in Shenzhen (China) described in Section 3. The dataset includes 29200 buildings, covering a total area of 187.53 km^2. The dataset contains labels of each individual buildings and therefore can be considered as the “ground-truth” for this study. The dataset was randomly divided into 3 sub-sets by considering the distributions of function categories: 17600 samples are used as the training set, 5800 samples as validation set, and the other 5800 as test set.

As in other classification studies, we made use of the following metrics to evaluate the performance of the proposed model: accuracy, Cohen's kappa coefficient, precision, and recall.

- **Accuracy**: It is the percentage of correctly inferred buildings out of the total number of buildings, and is an intuitive performance measure. It refers to all categories.
- **Cohen's kappa coefficient:** It is an accuracy metric normalized by the imbalance of classes in the data, by considering the possibility of correct identification occurring by chance. It compares the actual accuracy with the expected accuracy, which is the accuracy that results from classification by random chance. It refers to all categories.
- **Precision:** It is the percentage of correctly identified buildings out of the total number of buildings identified with that specific function category. For example, the precision of function category "Hotel" is the probability that a randomly selected building inferred as "Hotel" is actually a "Hotel". Each building function category has a precision value.
- **Recall:** It is the percentage of correctly inferred buildings out of the number of buildings of that specific function category. For example, the recall of function category "Hotel" is the probability that a randomly selected "Hotel" building was inferred as "Hotel". Each building function category has a recall value.

Five additional classification methods were implemented as benchmarks against the proposed model. Note that other state-of-the-art methods for inferring building functions require other datasets as input, such as street view images and remote sensing imagery, making it difficult to directly use these methods as baselines for the comparisons. Therefore, we only compare the performance of our proposed model with three deep learning models (Word2vec, Place2vec and Block2vec) and two conventional machine learning models (XGBoost and RF).

- **Word2vec**: This model was used by Zhai et al. (2019) as a benchmark to infer function of a region based on POIs. Briefly, based on the distance between POIs, k-nearest POIs around a center POI can be identified. Subsequently, a POI sequence is built according to such distances. Each POI sequence is viewed as a sentence and fed into Word2vec proposed by Mikolov et al. (2013a), in which the context vector of each POI category can be obtained. Meanwhile, the building vector can be specified mathematically by weighted average of the POI vector in that building according to different categories' POIs. Finally, as the feature, building vectors are fed into random forest (RF) to infer building function.
- **Place2vec**: This model was proposed by Zhai et al. (2019) to infer function of a region based on POIs. The way of building POI sequences is same as Word2vec. The only difference is that Place2vec considers the spatial information using nearest neighbor approach, which augments the spatial context of a region.
- **Block2vec:** This model was proposed by Sun et al. (2021) and only considers the POI within study entities to infer the function. Briefly, the POI sequence

is built according to the distance to the centroid of the building footprint. Here, the POI sequence is at a fixed length s. If the POI sequence in this building is less than s, the specific characters would be added. Conversely, the excess POIs would be eliminated. Additionally, the model also considers the 4 buildings closest to the target building. The whole model is built by a deep learning method, i.e., long-short term memory (LSTM). After constructing the vector of a building, random forest is applied to infer the building function.

- **XGBoost**: The eXtreme Gradient Boosting (XGBoost) algorithm is a popular tree-based method for classification tasks (Chen and Guestrin 2016). For XGBoost, we only consider the feature vectors of the target building itself to infer its building function category.
- **Random Forest (RF)**: It is a classic and popular machine learning model for classification tasks (Breiman 2001). RF is selected mainly because it is found to be easy to train, to have high performances and not to over-fit the data (Breiman 2001, Cutler, Cutler and Stevens 2012). For the RF model, we only consider the feature vectors of the target building itself to infer its building function category.

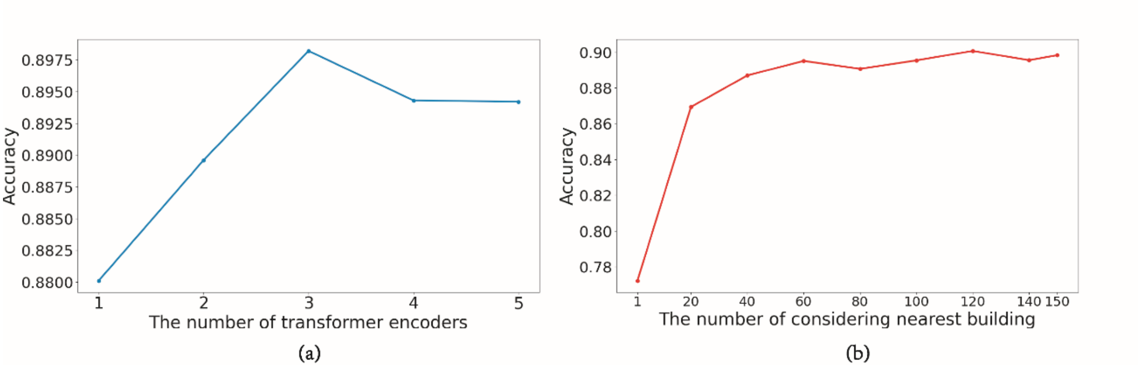
5.2 Tunning hyperparameters

There are four hyperparameters in the proposed model: the number of encoder layers l, the number of nearby buildings to be additionally considered n, the dimension of the refined vector of service facilities (see Section 4.3), and that of human activity pattern (See Section 4.3). The section mainly focuses on tunning these hyperparameters. Noted that the reported accuracies are computed on the validation set (instead of the test set).

Figure 7 demonstrates the different accuracies when l and n are varied. In Figure 7a, we fix the number of nearest buildings, the dimension of the refined vector of service facilities, and that of human activity pattern as 150 (i.e., $n = 150$), 9, and 25, respectively. The blue line shows how the accuracy changes with the number of transformer encoders. When the number of transformer encoders is 3, the model is more accurate than other values from 1 to 5. Thereby, we use 3 transformer encoders in our model (i.e., $l = 3$). In Figure 7b, we fix 3 transformer encoders and observe that the accuracies generally increase with the increase of the number of nearby buildings considered. This is expected as the more nearby buildings considered, the more "geographic context" information is added to the proposed model. When $n = 120$, the accuracy reaches the highest value of 0.899. Compared to only one nearby building being considered, considering 120 nearby buildings improves the accuracy by 16%.

Since the vector dimension of service facilities and human activity pattern are related to the number of multi-head attention mentioned in Section 4.5 (e.g., if the numbers of heads are 3 and 6, the vector dimensions are 3^2 and 6^2 respectively), we tune the vector dimension by changing the number of heads. We randomly fix the number of heads of human activity pattern as 5 and change that of service facilities from 2 to 7. In Figure 8a, the orange line shows that the performance is the best when the number of heads is 4 and 7, achieving 0.909. Considering the computational cost, we choose 4. In Figure 8b (the green line), we fix the number of heads of service facilities as 4 and change the one of human activity pattern from 2 to 7. The results demonstrate that when the number of heads equals 5, the performance is better, reaching 0.909.

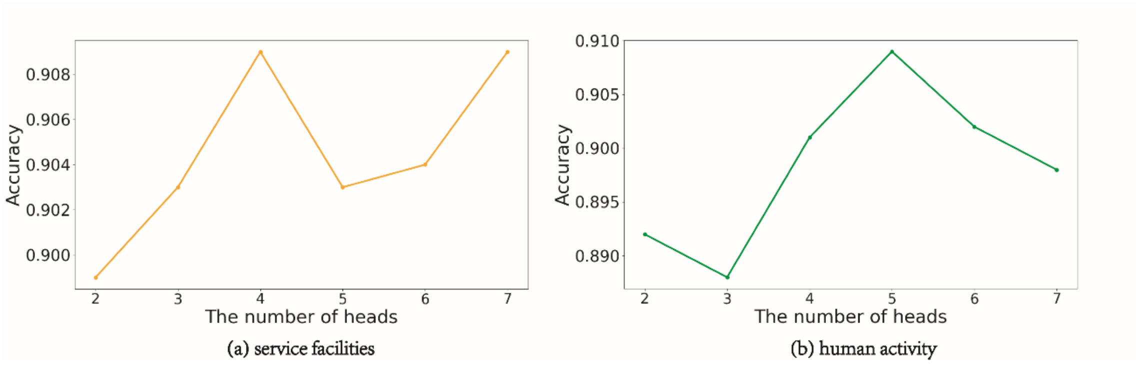
Considering the above results, for the follow-up analysis, we set the number of encoder layers to 3, the number of nearby buildings to 120, the number of heads in service facilities and human activity pattern to 4 and 5 respectively (i.e., the corresponding vector dimensions are 16 and 25 respectively).



(a)

(b)

Figure 7. Accuracies in different numbers of transformer encoders and nearest buildings.



(a) service facilities

(b) human activity

Figure 8. Accuracies in different numbers of heads in service facilities and human activity pattern.

5.3 Performance comparison

This section compares the performance of our proposed model against the five baselines. For each of the baselines, we chose the best model (in terms of predictive accuracy) after tuning its corresponding hyperparameters (considering the parameter settings of the original studies). All evaluation metrics are calculated on the test dataset. Table 3 illustrates that our proposed model outperforms the baseline models in both accuracy and kappa coefficient. Our model achieves the highest accuracy of 90.8% with the kappa coefficient of 0.87. XGBoost and RF perform relatively worse, because these models can only consider the features of the target building itself, and no nearby buildings are considered. This again demonstrates the importance of considering nearby buildings (i.e., "geographic context") for inferring the function of a target building. Additionally, Block2vec, Word2vec and Place2vec perform significantly worse. Note that these three methods were proposed to infer functions of larger spatial regions, e.g., traffic analysis zones or big street blocks (instead of individual buildings). Their poor performance is presumably due to the fact that many individual buildings do not contain any POIs, making it difficult to infer their functions.

Table 3. Comparison of the proposed model and baselines.

	Model					
Metric	Our model	XGBoost	RF	Block2vec	Word2vec	Place2vec
Accuracy	**0.908**	0.716	0.654	0.441	0.440	0.438
kappa	**0.87**	0.58	0.46	0.05	0.05	0.05

5.4 Ablation study

This section evaluates the importance of different components in our proposed model. All evaluation metrics are calculated on the test dataset. The variant “conventional transformer encoder” uses the same architecture of the proposed model, but the Geo-PEG part was replaced by simply assigning the positions (i.e., 0, 1, …, n) as the “ordering” of the buildings. Table 4 illustrates that our model achieves the best performance, followed by the conventional transformer encoder, transformer encoder with human mobility only, and transformer encoder with service facility only. This suggests that the proposed components, i.e., the Geo-PEG module, the service facility features, and the human mobility features, all contribute positively to the classification accuracies.

Table 4. Comparison of the proposed model with its own variants

	Model			
Metric	Our model	Conventional	Only service facility	Only human mobility

		transformer encoder		
Accuracy	**0.908**	0.895	0.883	0.886
kappa	**0.87**	0.85	0.84	0.84

5.5 Training dataset size

This section investigates the relationship between the classification accuracy and the amount of training samples. We train the proposed model on the training dataset (17600 samples) that increases by 20% from 20% to 100%, and evaluate the performance of the trained model on the test dataset. Table 5 demonstrates that the model's performance increases as the amount of training samples increases. Note that our model can reach a classification accuracy of 0.817 and a kappa coefficient of 0.75 when only 20% of the training samples (3500 samples) were used. This is desirable and shows that our model can achieve a decent performance even with a very limited number of training samples.

Table 5. Performance of our model on different amount of training samples.

	Amount of training samples				
Metric	20%	40%	60%	80%	100%
Accuracy	0.817	0.851	0.872	0.887	0.908
kappa	0.75	0.79	0.82	0.84	0.87

5.6 Analysis of the classification performance

5.6.1 Spatial distribution of the classification performance

To investigate which geographical factors influence the classification performance, we visualize the spatial distribution of classification accuracy, building density and building function diversity (Figure 9). Firstly, we divide the Nanshan district into different areas (DAs) using the road network downloaded from OpenStreetMap. Subsequently, for each DA, we calculate the classification accuracy, building density and building function diversity, which can be described as:

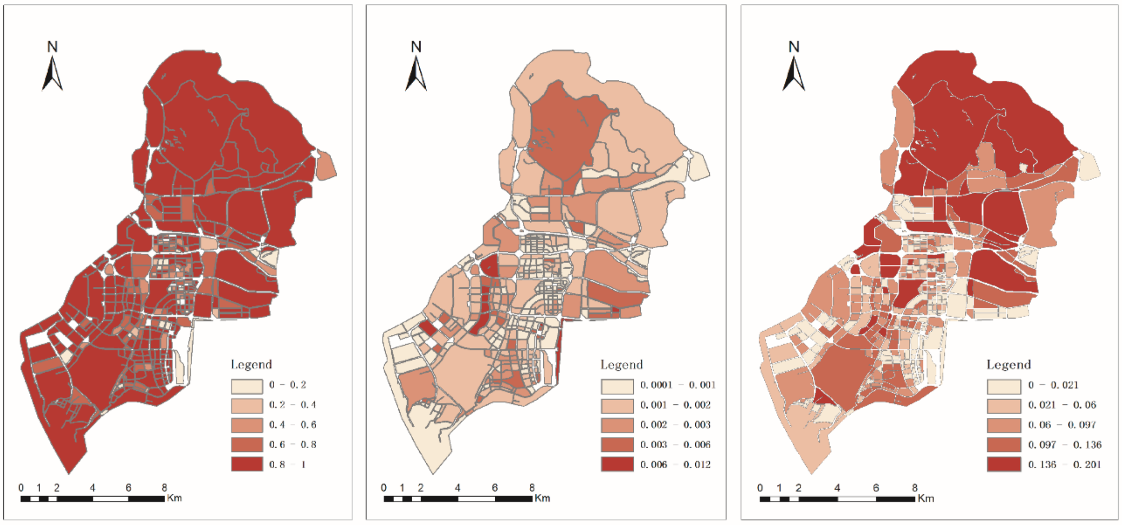
$$accuarcy_i = \frac{SI_i}{BN_i} \tag{21}$$

$$density_i = \frac{BN_i}{BA_i} \tag{22}$$

$$entropy_i = -\sum_c p(c)\, log_2 p(c) \tag{23}$$

Where i represents the i^{th} DA, SI denotes to the number of successfully inferred buildings, BN is the number of total buildings, and BA is the area sum of buildings. Additionally, Shannon entropy is used to measure the building function diversity in each DA, in which $p(c)$ indicates the proportion of c^{th} categories of building in all categories.

Figure 9a shows the accuracy of inferring building functions, which is divided into five groups relying on the equal interval method. The change from light orange to red indicates the increase of the classification accuracy. According to Figure 9a, the majority of the areas are red, indicating that the proposed model performs well regardless of the location of the area. Both Figure 9b and Figure 9c are classified into five groups using the natural breaks method. As can be seen, most of the areas with low classification accuracies are located in the areas with low building density.



(a) Accuracy for inferring building function (b) Building density (c) Building function diversity

Figure 9. Spatial distribution of classification accuracy, building density and function diversity.

5.6.2 Confusion matrix

To better understand the classification performance of different building categories, Table 6 shows the confusion matrix of building function classification of the proposed model. Here, RC denotes to the recall, PR refers to the precision. The high performance of the categories UV, UR, RT, CF, IP, and EDU indicates the model can predict these categories with high accuracy. However, our model has a relatively poor performance in the building categories CA and HT, owing to the limited number of samples of these categories.

Table 6. Confusion matrix of the building function classification.

		predicted													
		UV	UR	BO	CA	SC	HT	RT	CF	IP	AM	EDU	MD	Total	RC(%)
actual	UV	2415	19	0	0	0	0	0	7	4	4	2	0	2451	98.5
	UR	17	1417	11	0	0	0	1	9	16	3	5	0	1479	95.8
	BO	2	38	77	0	1	1	0	16	11	0	2	0	148	52
	CA	0	0	0	0	0	0	1	0	0	0	0	0	1	0
	SC	0	5	4	0	21	0	1	5	0	0	0	0	36	58.3
	HT	0	9	3	0	0	3	2	1	1	0	0	0	19	15.8
	RT	0	15	1	0	0	1	76	4	0	0	3	1	101	75.2
	CF	14	39	7	0	6	0	1	655	17	2	13	0	754	86.9
	IP	2	20	13	0	0	0	0	35	371	1	10	1	453	81.9
	AM	3	19	2	0	0	1	0	11	1	32	2	0	71	45.1
	EDU	4	45	3	0	1	0	1	6	7	2	183	0	252	72.6
	MD	4	11	1	0	0	0	0	1	0	0	1	17	35	48.6
	Total	2461	1637	122	0	29	6	83	750	428	44	221	19	5800	
	PR(%)	98.1	86.6	63.1	0	72.4	50	91.6	87.3	86.7	72.7	82.8	89.5		

5.7 Model applicability in another urban area

To evaluate the applicability of the proposed model in other urban areas, we select Luohu district, another district in Shenzhen city, as another test area. Luohu district is the old town of Shenzhen, while the study area of Nanshan district is the new town and central business district (CBD) of Shenzhen, which means that these two districts have very different built environment and spatial configuration patterns. This section uses 19200 building samples located in the Luohu district, wherein 4800 samples are randomly selected as the training set, and the other 14400 samples are selected as the test set to evaluate the performance of the proposed model. Table 7 shows the model performance on the Luohu district. Firstly, we directly utilize the trained model from the Nanshan district to evaluate performance on the test set, which obtains the accuracy of 66.2% with a kappa coefficient of 0.49. Subsequently, we randomly add 500 samples from the Luohu training set to further train the model, which achieves an accuracy of 75.32% with a kappa coefficient of 0.63 on the Luohu test set. Finally, we completely re-train the proposed model using only the Luohu training set and evaluate the trained model on the Luohu test set. The results show an accuracy of 81.67% and a kappa coefficient of 0.72. All these show that the proposed model has good applicability to other urban areas.

Table 7. Model applicability in Luohu.

Model	Accuracy	Kappa
Trained in Nanshan dataset	66.2%	0.49
Model trained in Nanshan dataset + 500 samples in Luohu	75.32%	0.63
Re-train model with 4800 samples of Luohu	81.67%	0.72

6. Discussions

The results of the performance comparison between the proposed model and the baselines illustrate that: 1) When inferring the function of a target building, it is important to consider its nearby buildings; Information regarding these nearby buildings helps to “contextualize” the target building. 2) Considering both features about the service facility and human mobility of buildings leads to an improvement of the classification accuracies. 3) Compared to the conventional position embedding method in natural language processing, the proposed geo-aware position embedding generator, which considers the spatial distances between buildings and spatial distribution around individual buildings, helps to further improve the model performance. In general, our proposed model achieves a classification accuracy of 90.8%, and a kappa coefficient of 0.87, which is better than all the baselines. This illustrates that the proposed model, i.e., the combination of Geo-PEG, transformer encoders, service facility features, and human mobility features, has a strong ability in inferring building functions.

Additionally, the performance of varying amounts of training samples illustrates that the amounts of training samples can influence the model performance. More importantly, the proposed model can achieve an accuracy of 81.7% with a kappa coefficient of 0.75 on only 20% of the training samples (3500 samples), which illustrates that our proposed model performs well even with a limited amount of training dataset. This is desirable, especially for the cases when it is difficult to obtain a large amount of labeled data. Meanwhile, by evaluating the proposed model at another urban area, we show that the proposed model is not specific to the study area and has a good transferability to other urban areas.

The evaluation results also show that the proposed model can successfully infer the building function at a high accuracy no matter where the building is located at. While the high accuracy is found in most places, there are few places with low accuracy. To better understand performance in detail, we visualize the distribution of building density and function diversity. The results show that the areas with low classification accuracy are often the areas with low building density, suggesting that there is limited "contextual" information for inferring the function of the target building in such areas. Meanwhile, according to the distribution of building function diversity, our model also can perform very well even if in high diversity areas.

Looking at the classification accuracies of individual building function categories, we found that the function categories of UV ("urban village”), UR ("urban residential”), RT ("recreation & tourism”), CF ("company & factories”), IP ("industrial park”), and EDU (“education”) can be recognized with high accuracy due to sufficient training samples. The categories of BO ("business office”) and AM ("administrative”) have relatively sufficient training samples but have relatively low accuracy. In terms of BO, around 20% of the BOs are misclassified as URs, presumably because the URs closed to CBD have similar architecture characteristics and functional configuration as BOs, leading to misclassification. Around 26% of the AMs are misclassified as URs, since most of the AMs are local community councils, which is normally surrounded by UR buildings.

Several limitations in this study should be mentioned. Firstly, while this study evaluated the model transferability, we could only evaluate the proposed model on the other district in the same city due to the limited datasets. It would be interesting to investigate how our model performs when applied to other types of cities. Secondly, in the geo-aware position embedding layer, this article uses the same parameter to control the range for every building, ignoring the spatial heterogeneity. This can be also improved in the future by setting the range of each building according to its geographic context (e.g., density of nearby buildings or even the bigger area to which the building belongs). Thirdly, when building the input features of a building, this study used Euclidean distances to measure its proximity to bus/metro stations and parks, mainly trying to reduce computational costs. To better reflect the proximity of buildings to these facilities, it might be better to employ network distance.

7. Conclusion

This paper proposed a novel geo-aware neural network model to infer functions of individual buildings, which makes use of the POI distributions, human mobility patterns of the target building and its nearby buildings. The model includes a geo-aware position embedding generator and transformer encoders to better capture the deep relationships between the target building and its nearby buildings (i.e., the "geographic context" of the target building). In summary, the evaluation results show that our proposed model significantly outperforms the baselines. Meanwhile, it performs well even when the training dataset is limited. It also has a good transferability for other urban areas. All these results demonstrate that the proposed model is an effective and reliable method to infer building function, which is important for city management and policy making. Further research should aim to improve the model by considering spatial heterogeneity, which might help to further enhance the model applicability in other cities around the world. Meanwhile, it might be useful to integrate features from remote sensing images and street-view images to further improve the classification performance of the proposed model.

Acknowledgement

Declarations of interest

None

Reference

Breiman, L. (2001) Random Forests. *Machine Learning,* 45**,** 5-32.

Cao, R., W. Tu, C. Yang, Q. Li, J. Liu, J. Zhu, Q. Zhang, Q. Li & G. Qiu (2020) Deep learning-based remote and social sensing data fusion for urban region function recognition. *ISPRS Journal of Photogrammetry and Remote Sensing,* 163**,** 82-97.

Chen, T. & C. Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. San Francisco, California, USA: Association for Computing Machinery.

Chen, Y., X. Liu, X. Li, X. Liu, Y. Yao, G. Hu, X. Xu & F. Pei (2017) Delineating urban functional areas with building-level social media data: A dynamic time warping

(DTW) distance based k-medoids method. *Landscape and Urban Planning,* 160**,** 48-60.

Cutler, A., D. R. Cutler & J. R. Stevens. 2012. Random Forests. In *Ensemble Machine Learning: Methods and Applications,* eds. C. Zhang & Y. Ma, 157-175. Boston, MA: Springer US.

Deng, Y., R. Chen, J. Yang, Y. Li, H. Jiang, W. Liao & M. Sun (2022) Identify urban building functions with multisource data: a case study in Guangzhou, China. *International Journal of Geographical Information Science***,** 1-26.

Gong, P., B. Chen, X. Li, H. Liu, J. Wang, Y. Bai, J. Chen, X. Chen, L. Fang, S. Feng, Y. Feng, Y. Gong, H. Gu, H. Huang, X. Huang, H. Jiao, Y. Kang, G. Lei, A. Li, X. Li, X. Li, Y. Li, Z. Li, Z. Li, C. Liu, C. Liu, M. Liu, S. Liu, W. Mao, C. Miao, H. Ni, Q. Pan, S. Qi, Z. Ren, Z. Shan, S. Shen, M. Shi, Y. Song, M. Su, H. Ping Suen, B. Sun, F. Sun, J. Sun, L. Sun, W. Sun, T. Tian, X. Tong, Y. Tseng, Y. Tu, H. Wang, L. Wang, X. Wang, Z. Wang, T. Wu, Y. Xie, J. Yang, J. Yang, M. Yuan, W. Yue, H. Zeng, K. Zhang, N. Zhang, T. Zhang, Y. Zhang, F. Zhao, Y. Zheng, Q. Zhou, N. Clinton, Z. Zhu & B. Xu (2020) Mapping essential urban land use categories in China (EULUC-China): preliminary results for 2018. *Science Bulletin,* 65**,** 182-187.

Haberl, H., M. Wackernagel & T. Wrbka (2004) Land use and sustainability indicators. An introduction. *Land Use Policy,* 21**,** 193-198.

He, J., X. Li, P. Liu, X. Wu, J. Zhang, D. Zhang, X. Liu & Y. Yao (2021) Accurate Estimation of the Proportion of Mixed Land Use at the Street-Block Level by Integrating High Spatial Resolution Images and Geospatial Big Data. *IEEE Transactions on Geoscience and Remote Sensing,* 59**,** 6357-6370.

Hu, S., Z. He, L. Wu, L. Yin, Y. Xu & H. Cui (2020) A framework for extracting urban functional regions based on multiprototype word embeddings using points-of-interest data. *Computers, Environment and Urban Systems,* 80**,** 101442.

Hu, S. G. & L. Wang (2013) Automated urban land-use classification with remote sensing. *International Journal of Remote Sensing,* 34**,** 790-803.

Huang, B., B. Zhao & Y. Song (2018) Urban land-use mapping using a deep convolutional neural network with high spatial resolution multispectral remote sensing imagery. *Remote Sensing of Environment,* 214**,** 73-86.

Huang, H., X. A. Yao, J. M. Krisp & B. Jiang (2021) Analytics of location-based big data for smart cities: Opportunities, challenges, and future directions. *Computers, Environment and Urban Systems,* 90**,** 101712.

Ibraeva, A., B. Van Wee, G. H. d. A. Correia & A. Pais Antunes (2021) Longitudinal macro-analysis of car-use changes resulting from a TOD-type project: The case of Metro do Porto (Portugal). *Journal of Transport Geography,* 92**,** 103036.

Li, Y. & T. Yang. 2018. Word embedding for understanding natural language: a survey. In *Guide to big data applications*, 83-104. Springer.

Liu, S., C. Zhou, J. Rong, Y. Bian & Y. Wang (2022) Concordance between Regional Functions and Mobility Features Using Bike-sharing and Land-use Data near Metro Stations. *Sustainable Cities and Society,* 84**,** 104010.

Liu, X., C. Kang, L. Gong & Y. Liu (2016) Incorporating spatial interaction patterns in classifying and understanding urban land use. *International Journal of Geographical Information Science,* 30**,** 334-350.

Liu, X. & Y. Long (2015) Automated identification and characterization of parcels with OpenStreetMap and points of interest. *Environment and Planning B: Planning and Design,* 43**,** 341-360.

Liu, X. P., N. Niu, X. J. Liu, H. Jin, J. P. Ou, L. M. Jiao & Y. L. Liu (2018) Characterizing mixed-use buildings based on multi-source big data. *International Journal of Geographical Information Science,* 32**,** 738-756.

MacKillop, F. (2012) Climatic city: Two centuries of urban planning and climate science in Manchester (UK) and its region. *Cities,* 29**,** 244-251.

Mikolov, T., K. Chen, G. Corrado & J. Dean (2013a) Efficient estimation of word representations in vector space. *preprint arXiv:1301.3781.*

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado & J. Dean (2013b) Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems,* 26.

Niu, H. & E. A. Silva (2021) Delineating urban functional use from points of interest data with neural network embedding: A case study in Greater London. *Computers, Environment and Urban Systems,* 88**,** 101651.

Niu, N., X. P. Liu, H. Jin, X. Y. Ye, Y. Liu, X. Li, Y. M. Chen & S. Y. Li (2017) Integrating multi-source big data to infer building functions. *International Journal of Geographical Information Science,* 31**,** 1871-1890.

Pacifici, F., M. Chini & W. J. Emery (2009) A neural network approach using multi-scale textural metrics from very high-resolution panchromatic imagery for urban land-use classification. *Remote Sensing of Environment,* 113**,** 1276-1292.

Pei, T., S. Sobolevsky, C. Ratti, S.-L. Shaw, T. Li & C. Zhou (2014) A new insight into land use classification based on aggregated mobile phone data. *International Journal of Geographical Information Science,* 28**,** 1988-2007.

Rahnama, M. R. (2021) Forecasting land-use changes in Mashhad Metropolitan area using Cellular Automata and Markov chain model for 2016-2030. *Sustainable Cities and Society,* 64**,** 102548.

Srivastava, S., J. E. V. Munoz, S. Lobry & D. Tuia (2020) Fine-grained landuse characterization using ground-based pictures: a deep learning solution based on globally available data. *International Journal of Geographical Information Science,* 34**,** 1117-1136.

Srivastava, S., J. E. Vargas-Munoz, D. Swinkels & D. Tuia. 2018. Multi-label Building Functions Classification from Ground Pictures using Convolutional Neural Networks. In *2nd ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery (GeoAI)*, 43-46. Seattle, WA.

Srivastava, S., J. E. Vargas-Muñoz & D. Tuia (2019) Understanding urban landuse from the above and ground perspectives: A deep learning, multimodal solution. *Remote Sensing of Environment,* 228**,** 129-143.

Sun, Z., H. Jiao, H. Wu, Z. Peng & L. Liu (2021) Block2vec: An Approach for Identifying Urban Functional Regions by Integrating Sentence Embedding Model and Points of Interest. *ISPRS International Journal of Geo-Information,* 10.

Tong, X.-Y., G.-S. Xia, Q. Lu, H. Shen, S. Li, S. You & L. Zhang (2020) Land-cover classification with high-resolution remote sensing images using transferable deep models. *Remote Sensing of Environment,* 237**,** 111322.

Tu, W., J. Z. Cao, Y. Yue, S. L. Shaw, M. Zhou, Z. S. Wang, X. M. Chang, Y. Xu & Q. Q. Li (2017) Coupling mobile phone and social media data: a new approach to understanding urban functions and diurnal patterns. *International Journal of Geographical Information Science,* 31**,** 2331-2358.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser & I. Polosukhin (2017) Attention is all you need. *Advances in neural information processing systems,* 30.

Wang, Z. & V. Moosavi. 2020. From PIace2Vec to Multi-Scale Built-Environment Representation: A General-Purpose Distributional Embedding for Urban Data Analysis. In *Proceedings of the 4th ACM SIGSPATIAL Workshop on Location-Based Recommendations, Geosocial Networks, and Geoadvertising*, Article 1. Association for Computing Machinery.

Xiao, T., P. Xu, R. Ding & Z. Chen (2022) An interpretable method for identifying mislabeled commercial building based on temporal feature extraction and ensemble classifier. *Sustainable Cities and Society,* 78**,** 103635.

Yan, B., K. Janowicz, G. Mai & S. 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*, Article 35. Redondo Beach, CA, USA: Association for Computing Machinery.

Yao, Y., X. Li, X. Liu, P. Liu, Z. Liang, J. Zhang & K. Mai (2017) Sensing spatial distribution of urban land use by integrating points-of-interest and Google Word2Vec model. *International Journal of Geographical Information Science,* 31**,** 825-848.

Zhai, W., X. Bai, Y. Shi, Y. Han, Z.-R. Peng & C. Gu (2019) Beyond Word2vec: An approach for urban functional region extraction and identification by combining Place2vec and POIs. *Computers, Environment and Urban Systems,* 74**,** 1-12.

Zhang, J. B., X. Li, Y. Yao, Y. Hong, J. Y. He, Z. W. Jiang & J. C. Sun (2021) The Traj2Vec model to quantify residents' spatial trajectories and estimate the proportions of urban land-use types. *International Journal of Geographical Information Science,* 35**,** 193-211.

Zhang, X., Y. Sun, A. Zheng & Y. Wang (2020) A New Approach to Refining Land Use Types: Predicting Point-of-Interest Categories Using Weibo Check-in Data. *ISPRS International Journal of Geo-Information,* 9**,** 124.

Zhong, C., X. Huang, S. Müller Arisona, G. Schmitt & M. Batty (2014) Inferring building functions from a probabilistic model using public transportation data. *Computers, Environment and Urban Systems,* 48**,** 124-137.

Zhuo, L., Q. L. Shi, C. Y. Zhang, Q. P. Li & H. Y. Tao (2019) Identifying Building Functions from the Spatiotemporal Population Density and the Interactions of People among Buildings. *Isprs International Journal of Geo-Information,* 8.

Zong, L., S. He, J. Lian, Q. Bie, X. Wang, J. Dong & Y. Xie (2020) Detailed Mapping of Urban Land Use Based on Multi-Source Data: A Case Study of Lanzhou. *Remote Sensing,* 12.