Human-Instructed Deep Hierarchical Generative Learning for Automated Urban Planning

Dongjie Wang[1], Lingfei Wu[2], Denghui Zhang[3], Jingbo Zhou[4], Leilei Sun[5], Yanjie Fu[1]*

[1] University of Central Florida
[2] Pinterest
[3] Rutgers University
[4] Baidu Research
[5] Beihang University
wangdongjie@knights.ucf.edu, lwu@email.wm.edu, dhzhangai@gamil.com,
zhoujingbo@baidu.com, leileisun@buaa.edu.cn, yanjie.fu@ucf.edu

Abstract

The essential task of urban planning is to generate the optimal land-use configuration of a target area. However, traditional urban planning is time-consuming and labor-intensive. Deep generative learning gives us hope that we can automate this planning process and come up with the ideal urban plans. While remarkable achievements have been obtained, they have exhibited limitations in lacking awareness of: 1) the hierarchical dependencies between functional zones and spatial grids; 2) the peer dependencies among functional zones; and 3) human regulations to ensure the usability of generated configurations. To address these limitations, we develop a novel human-instructed deep hierarchical generative model. We rethink the urban planning generative task from a unique functionality perspective, where we summarize planning requirements into different functionality projections for better urban plan generation. To this end, we develop a three-stage generation process from a target area to zones to grids. The first stage is to label the grids of a target area with latent functionalities to discover functional zones. The second stage is to perceive the planning requirements to form urban functionality projections. We propose a novel module: functionalizer to project the embedding of human instructions and geospatial contexts to the zone-level plan to obtain such projections. Each projection includes the information of land-use portfolios and the structural dependencies across spatial grids in terms of a specific urban function. The third stage is to leverage multi-attentions to model the zone-zone peer dependencies of the functionality projections to generate grid-level land-use configurations. Finally, we present extensive experiments to demonstrate the effectiveness of our framework.

Introduction

Urban planning is vital for building up a sustainable and vigorous community. As a complicated and time-consuming task, traditional practice heavily depends on experts' personal experiences. The variance among urban planners may result in biases and implausible solutions. Thanks to the explosive development of deep learning and internet-of-things, the handful of methodologies and ubiquitously available geo-social, urban and mobile data provide us with a new data-driven perspective to re-investigate urban planning.

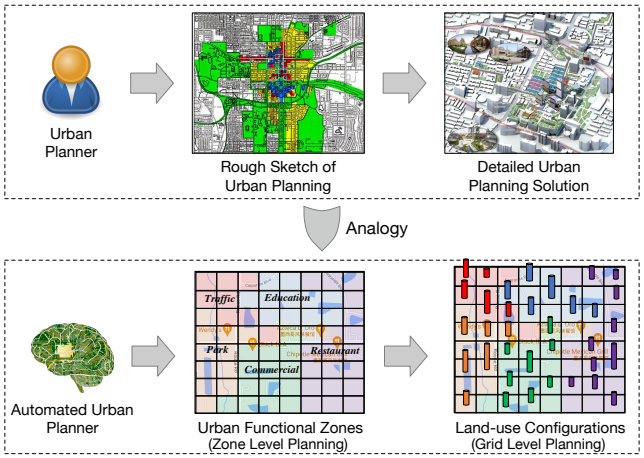


Figure 1: The automated urban planner can mimic the workflow of urban experts by first generating zone-level planning and then refining it to grid-level planning.

There are considerable existing works related to automated urban planning (Wang et al. 2021a; Shen et al. 2020; Ye, Du, and Ye 2021; Wang et al. 2021b). For example, motivated by the remarkable success of deep image generation, (Wang et al. 2020) proposes a land-use configuration generation framework, namely LUCGAN, which can generate a land-use configuration automatically for an empty geographical area based on surrounding contexts. While the existing works have achieved promising results, there are still several limitations: 1) hierarchical relationships between high-level urban functional zones and the detailed urban planning scheme are ignored; 2)mutual dependencies and influences among the planning of different subareas are omitted. 3) human instructions from planning experts, such as safety level, greening rate, volume rate, and etc, cannot be perceived by model;

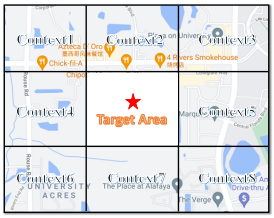
Therefore, in this paper, we study the research problem of how to employ deep models to make automated urban plan-

*Contact Author

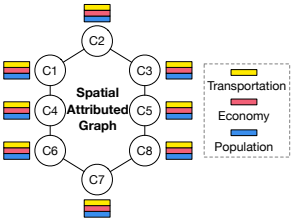
ning more intelligent. To settle the problem, we can formulate urban planning as a deep conditional generative task, in which human instructions and surrounding contexts can be regarded as the generative condition, and spatial hierarchical relationships and planning dependencies can be considered as the generative constraints. The objective is to generate an urban solution constrained by various factors.

However, there are three unique challenges in the defined generative task: 1) **Challenge 1: Capturing Spatial Hierarchical Relations**: urban functional zones reflect the land-use layout of a geographical area, which provides the bedrock of a land-use configuration. Neglecting such spatial hierarchical relations between urban functional zones and the land-use configuration may result in the unstable generation performance. But how can we model the spatial hierarchies during the generation process? 2) **Challenge 2: Capturing Planning Dependency Among Subareas**: planning solutions of different sub geographical areas are mutually dependent and affected. In a geographical area, for example, if some subareas have been built up with a lot of business buildings, the other subareas will be built up with more entertainment as a supplement to the urban functions. But how can we capture the planning dependencies among different subareas? 3) **Challenge 3: Integrating Human instructions from Planning Experts**: urban planning is a highly complicated and personalized task. To produce plausible urban planning results, planning experts always consider various realistic factors (*e.g.* greening rate level, safe level, volume rate level). But how can we integrate such human instructions for improving the personalized generative capability of the model?

To tackle the above challenges, we propose a novel Human-**I**nstructed Deep **H**ierarchical Generative Framework (**IHPlanner**), which can generate a desired land-use configuration for an empty area based on human instructions and surrounding contexts, as well as considering the spatial hierarchies and the planning dependencies. Our main contributions can be summarized as follows: 1) **Formulating the automated urban planning as a multi-scale generation framework**. The classical workflow of urban experts is to first design a rough sketch, then fill concrete designing elements to obtain the final urban plan. Imitating such a designing workflow, the proposed multi-scale generation framework generates the coarse-grained skeleton (urban functional zones) at the first stage, and then produces the fine-grained urban plan (land-use configuration) based on the skeleton at the second stage. This framework setting automatically captures the spatial hierarchies between urban functional zones and land-use configurations. 2) **Involving human instructions from planning experts via conditional embedding**. We formulate the human instructions from experts as the generative condition in urban plan generation. To make our model perceive these conditions, we convert them into embedding vectors. To control the generation process, we concatenate such embedding vectors and regard them as the model input. 3) **Semantic segmentation-based generation to capture planning dependency**. Human instructions and surrounding contexts contain enormous semantics that implicitly reflect the planning requirements for

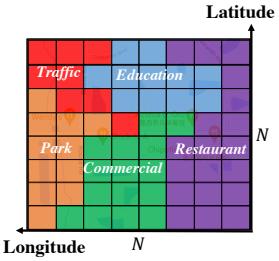


(a) Geospatial contexts encircle the target area from different directions.

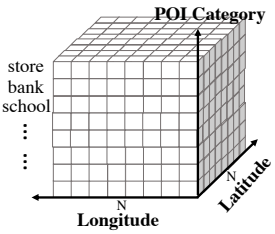


(b) The spatial attributed graph contains all features of geospatial contexts.

Figure 2: Illustration of target area and geospatial contexts.



(a) Zone-level planning is a 2-D matrix, which provides a high-level guidance for grid-level planning.



(b) Grid-level planning is represented by a 3-D tensor where we reserve the 3rd dimension for POI as each grid may contain multiple POI categories.

Figure 3: Urban functional zone and land-use configuration.

coarse-grained urban functional zones. Therefore, we design a planning semantic segmentation module to allocate the corresponding semantics to each urban functional zone respectively. We exploit the multi-head attention mechanism (Vaswani et al. 2017) to capture the dependencies among segmented semantics for quantifying the planning dependencies among subareas. Moreover, self-designed planning layers are developed to generate the final land-use configuration. 4) **Extensive experiments and case studies to validate the effectiveness of our framework**. We conduct all experiments and case studies based on the geographical data, traffic flow, road map, POIs, and check-in records of Beijing. We compare our proposed framework with six state-of-the-art deep generative models, and provide visualization to show the superiority of our framework.

Preliminaries

Definitions

Target Area and Geospatial Contexts. Target area is an empty and square geographical region (*e.g.* a square with a side length of 1 kilometer). Geospatial contexts are the surrounding environments, each of which has the same shape as the target area. Figure 2(a) illustrates that geospatial contexts encircle the target area from different directions. To leverage the information of geospatial contexts (Wang et al. 2020), we formulate such contexts as a spatial attributed graph de-

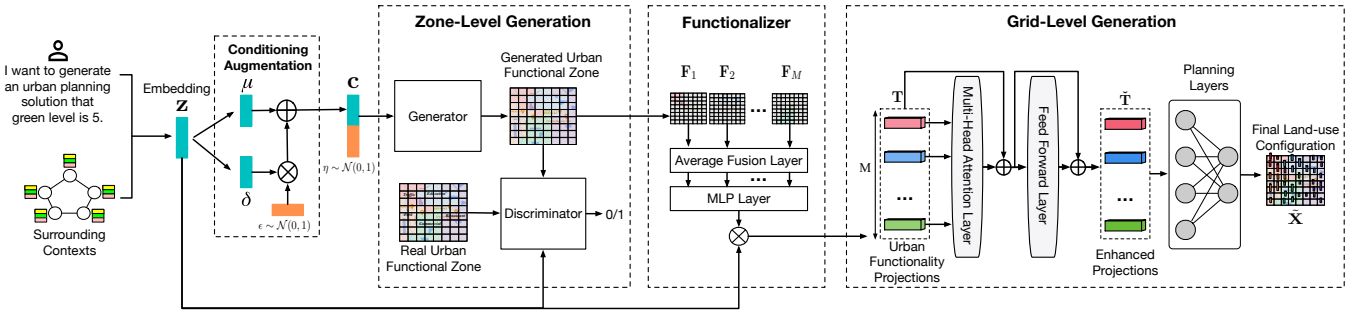


Figure 4: The overview of IHPlanner. It consists of four main steps: Conditioning Augmentation, Zone-level Generation, Functionalizer, and Grid-level Generation.

scribed in Figure 2(b). In this graph, a node is a geographical region, an edge reflects the spatial connectivity between any two regions, and the attributes of a node are the socioeconomic features of the corresponding region.

Urban Functional Zone and Land-use Configuration. In this paper, urban functional zones (*i.e.* zone-level planning) provide a planning foundation for the land-use configuration (*i.e.* grid-level planning). **For the urban functional zone**, following the idea in studies (Yuan et al. 2014), we utilize the geographical data and human mobility to extract. Specifically, we first divide a geographical area into $N \times N$ grids. Then, we consider the grids to be words and the human trajectories to be sentences, and so all trajectories inside the area constitute a document. Next, we use a topic model to discover the specific urban function label of each grid to obtain the final results. Figure 3(a) shows the data structure of such zones. The zone-level planning is a matrix, denoted by $\mathbf{U} \in \mathbb{R}^{N \times N}$, in which multiple grids affiliate to one urban function label. **For the land-use configuration**, we adopt the quantitative definition in studies (Wang et al. 2020). Specifically, we first split a geographical area into $N \times N$ grids. Then, we count how many POI (*i.e.* Point of Interest) locate in each grid under different POI categories. After that, we stack these counted results together as the final configuration. Figure 3(b) shows the data structure of such configuration, which is a tensor consisting of longitude, latitude, and POI category dimensions. The tensor is denoted by $\widehat{\mathbf{X}} \in \mathbb{R}^{N \times N \times C}$, where C is the number of POI categories.

Human Instruction. In this paper, human instruction is to guide the generation process of our planning framework. To allow our model to perceive such instruction, we quantify its semantic meaning into different levels. For instance, the range of green rate (*i.e.* the coverage of green plants of a geographical area) is $[0 \sim 1]$. We divide the green rate into multiple green rate levels. The label of these green rate levels is human instruction.

Problem Statement

Our goal is to develop an automated urban planner, which can generate a land-use configuration for an empty target area based on human instructions and geospatial contexts. Formally, given geospatial contexts denoted by $\mathcal{G}$, human instructions denoted by I, land-use configurations denoted by $\widehat{\mathbf{X}}$, we aim to find a mapping function $f : (\mathcal{G}, I) \rightarrow \widehat{\mathbf{X}}$. The function f takes geospatial contexts $\mathcal{G}$ and human instructions I as input, and outputs the corresponding grid-level land-use configuration $\widehat{\mathbf{X}}$.

Methodology

Framework Overview

Figure 4 shows the overview of our framework IHPlanner. The pipeline framework has four key components: **conditioning augmentation, zone-level generation, functionalizer, and grid-level generation**. Specifically, for an empty target area, we first preserve the planning requirements contained in human instructions and geospatial contexts into an embedding vector. Then, considering the data sparsity issue, we utilize the conditioning augmentation module to increase the data diversity. Next, we employ the zone-level generation module to generate the zone-level planning that provides a planning foundation for the grid-level generation. After that, in the functionalizer module, we project the semantics of planning requirements into different functional zones to obtain the urban functionality projections. This projection process converts the planning dependencies across functional zones into semantic correlations among these projections. Finally, in the grid-level generation module, we use multi-attentions to capture such semantic correlations, then employ planning layers to generate the grid-level planning.

Conditioning Augmentation

The dataset for automated urban planning is sparse, resulting in model overfitting or terrible generation performance of IHPlanner. We adopt the conditioning augmentation module to mitigate the learning issue. To make our method comprehend the planning semantics included in human instructions and surrounding geospatial contexts, we first convert the spatial attributed graph extracted from geospatial contexts into a graph embedding by (Kipf and Welling 2016; Wu et al. 2021, 2022), and then concatenate it with the one-hot vector of human instructions as the model input.

To be convenient, we adopt the k-th empty target area to explain the following calculation process. Specifically, we denote $\mathbf{z}^{(k)} \in \mathbb{R}^{1 \times O}$ as the concatenated embedding of human instructions and geospatial contexts, where O is

the size of the feature dimension. We first utilize the conditioning augmentation module to estimate the distribution of $\mathbf{z}^{(k)}$. Then, we randomly sample an augmented embedding $\mathbf{c}^{(k)}$ from the distribution, and regard it as the input of the zone-level generation module. The prior format of the estimated distribution is a normal distribution, denoted by $\mathcal{N}(\mu(\mathbf{z}^{(k)}), \delta(\mathbf{z}^{(k)}))$, where $\mu(.)$ and $\delta(.)$ indicate the mean and covariance function respectively. The mean and covariance value of the distribution are updated over learning process. We adopt the reparameterization technique to imitate the sampling operation, which can be formulated as follows:

$$\mathbf{c}^{(k)} = \boldsymbol{\mu}(\mathbf{z}^{(k)}) + \boldsymbol{\delta}(\mathbf{z}^{(k)}) \times \epsilon \tag{1}$$

where ϵ is a random variable vector sampled from a standard normal distribution $\mathcal{N}(0, 1)$.

Zone-level Generation

Inspired by the workflow of human planners, we can first generate a rough sketch of urban planning, then refine the sketch to the grid-level land-use configuration. Specifically, for the k-th empty target area, we first concatenate the embedding $\mathbf{c}^{(k)}$ and the random variable embedding $\boldsymbol{\eta}^{(k)}$ together, then input it into a generator to generate the urban functional zones. Here, $\boldsymbol{\eta}^{(k)}$ is sampled from the standard normal distribution $\mathcal{N}(0, 1)$, which can improve the robustness and generalization of model. Next, we combine the generated result and the embedding $\mathbf{z}^{(k)}$ together, then input it into a discriminator. The discriminator is to justify whether the input is the combination of the real urban functional zones $\mathbf{U}^{(k)}$ and $\mathbf{z}^{(k)}$. We alternatively optimize the generator and the discriminator until model convergence.

When optimizing the generator, we minimize equation 2:

$$\begin{aligned}\mathcal{L}_G = \sum_{k=1}^{K} log(1 - D(G(\boldsymbol{\eta}^{(k)}, \mathbf{c}^{(k)}), \mathbf{z}^{(k)})) \\ + \lambda \cdot KL[\mathcal{N}(\mu(\mathbf{z}^{(k)}), \delta(\mathbf{z}^{(k)}))||\mathcal{N}(0, 1)],\end{aligned} \tag{2}$$

where $KL[.]$ indicates the Kullback-Leibler (KL) divergence between the distribution $\mathcal{N}(\mu(\mathbf{z}^{(k)}), \delta(\mathbf{z}^{(k)}))$ and a standard normal distribution $\mathcal{N}(0, 1)$; λ is a scalar, which adjusts the contribution of the item $KL[.]$ in $\mathcal{L}_G$. $\mathcal{L}_G$ can be divided into two parts by "+". Intuitively, the first part tries to minimize the differences between the generated zone-level planning and the real zone-level planning, which improves the generation performance of the generator gradually. The second part tries to smooth the distribution $\mathcal{N}(\mu(\mathbf{z}^{(k)}), \delta(\mathbf{z}^{(k)}))$ produced by the conditioning augmentation module, which improves the diversity and quality of the input embedding $\mathbf{c}^{k}$ (Larsen et al. 2016).

When optimizing the discriminator, we maximize equation 3:

$$\begin{aligned}\mathcal{L}_D = \sum_{k=1}^{K} log(1 - D(G(\boldsymbol{\eta}^{(k)}, \mathbf{c}^{(k)}), \mathbf{z}^{(k)})) \\ + logD(\mathbf{U}^{(k)}, \mathbf{z}^{(k)}).\end{aligned} \tag{3}$$

Intuitively, $\mathcal{L}_D$ improves the discrimination ability by urging the discriminator to provide lower scores for the generated results and evaluate higher scores for the real standards.

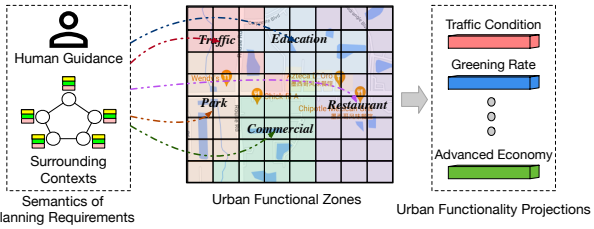


Figure 5: The information of planning requirements is projected into different urban functional zones to form urban functionality projections.

Ultimately, the well-trained generator can generate a suitable rough sketch of urban planning $\check{\mathbf{U}}^{(k)} \in \mathbb{R}^{N \times N}$ for the k-th empty area according to human instructions and geospatial contexts. Each value in $\check{\mathbf{U}}^{(k)}$ indicates the urban functionality label of the associated geographical location.

Functionalizer

An outstanding urban plan can be summarized by a few handfuls of urban functionalities such as convenient transportation, high green rate, and developed economy. In other words, to produce such an urban plan, our planning model should consider the planning requirements on these urban functionality sides. Thus, as illustrated in Figure 5, we project the planning requirements contained in human instructions and geospatial contexts into different functional zones to obtain urban functionality projections. This projection process lays the cornerstone for capturing the planning dependencies across different functional zones.

Specifically, for the k-th empty target area, we have generated the zone-level planning $\check{\mathbf{U}}^{(k)}$. Then, we divide the area into M zones according to the urban function labels in $\check{\mathbf{U}}^{(k)}$, denoted by $\mathbf{F}^{(k)} = [\mathbf{F}_1^{(k)}, \mathbf{F}_2^{(k)}, \cdots, \mathbf{F}_M^{(k)}]$, and $\mathbf{F}^{(k)} \in \mathbb{R}^{M \times N \times N}$. Next, we calculate the semantic proportion of planning requirements for each functional zone. After that, we multiply $\mathbf{z}^{(k)}$ with these semantic proportions to obtain urban functionality projections. The projection process can be formulated as follows:

$$\mathbf{T}^{(k)} = \text{Softmax}(\text{AVG\_Fusion}(\mathbf{F}^{(k)}) \cdot \mathbf{W}_a) \cdot \mathbf{z}^{(k)}, \tag{4}$$

where AVG_Fusion(.) column-wisely averages the information of each functional zone respectively, which changes the shape of $\mathbf{F}^{(k)}$ to $\mathbb{R}^{M \times N}$; $\mathbf{W}_a \in \mathbb{R}^{N \times 1}$ is the weight matrix; Softmax(.) outputs the semantic proportion value; $\mathbf{T}^{(k)} \in \mathbb{R}^{M \times O}$ are the final urban functionality projections, which implicitly reflect the planning requirements under different urban functionalities.

Grid-level Generation

Urban infrastructures and buildings in different functional zones are mutually dependent. For instance, if several functional zones have been planned with many commercialized buildings, planners will not put the same buildings in the nearby zones but instead add entertainment facilities to increase urban vibrancy. To capture such dependencies, we ap-

ply the multi-attentions (Vaswani et al. 2017) on urban functionality projections to obtain enhanced projections. Then, we input these enhanced projections into planning layers to produce the land-use configuration.

Specifically, for the k-th empty area, we input the urban functionality projections $\mathbf{T}^{(k)}$ into a multi-head attention layer to calculate the attention weight matrix. The multi-head attention layer consists of h single scaled dot-product attention layers. For a single attention layer, the calculation process is as follows:

$$\mathbf{A} = \text{Softmax}(\frac{\mathbf{Q} \cdot \mathbf{K}^T)}{\sqrt{d_k}}) \cdot \mathbf{V}, \tag{5}$$

where $\mathbf{A} \in \mathbb{R}^{M \times O}$ is the attention matrix; $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ are the query, key, and value matrix respectively. The three matrices all come from $\mathbf{T}^{(k)}$; d_k is the scaling factor; $\mathbf{Q} \cdot \mathbf{K}^T \in \mathbb{R}^{M \times M}$, which indicates the semantic similarity between any two of urban functionalities; These h single attention layers have different $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ matrices. These layers extract features from different semantic representation subspaces. Then we collect the attention weights of h layers together and add $\mathbf{T}^{(k)}$ to obtain $\mathbf{T}'^{(k)} \in \mathbb{R}^{M \times O}$,

$$\mathbf{T}'^{(k)} = \mathbf{T}^{(k)} + \text{Concat}(\mathbf{A}_1^{(k)}, \mathbf{A}_2^{(k)}, \cdots, \mathbf{A}_h^{(k)}) \cdot \mathbf{W}_T, \tag{6}$$

where $\mathbf{W}_T \in \mathbb{R}^{hO \times O}$ is the projection weight matrix. After that, we utilize a fully connected feed-forward network constituted by two linear layers to attain the enhanced projections $\check{\mathbf{T}}^{(k)} \in \mathbb{R}^{M \times O}$,

$$\check{\mathbf{T}}^{(k)} = \mathbf{T}'^{(k)} + \text{Relu}(\mathbf{T}'^{(k)} \cdot \mathbf{W}_1) \cdot \mathbf{W}_2, \tag{7}$$

where $\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{O \times O}$ are two weight matrices and Relu denotes the nonlinear transformation function. Next, we input $\check{\mathbf{T}}^{(k)}$ into planning layers to generate the final land-use configuration $\tilde{\mathbf{X}}^{(k)} \in \mathbb{R}^{N \times N \times C}$. This process can be formulated as,

$$\tilde{\mathbf{X}}^{(k)} = \mathbf{W}_u \cdot \check{\mathbf{T}}^{(k)} \cdot \mathbf{W}_d + \mathbf{b}, \tag{8}$$

where $\mathbf{W}_u \in \mathbf{R}^{N \times M}, \mathbf{W}_d \in \mathbb{R}^{O \times (N \times C)}$ are the weight matrices. During the generation process, $\mathbf{W}_u$ aims to consider the correlations among different enhanced projections; $\mathbf{W}_d$ aims to exploit the dependencies among different latent dimensions in these enhanced projections. $\mathbf{b} \in \mathbb{R}^{N \times (N \times C)}$ is the bias term. For the optimization, we minimize the differences between the real land-use configurations and the generated land-use configurations, the optimization objective is as follows:

$$\mathcal{L}_S = \sum_{k=1}^{K} ||\widehat{\mathbf{X}}^{(k)} - \tilde{\mathbf{X}}^{(k)}||^2 \tag{9}$$

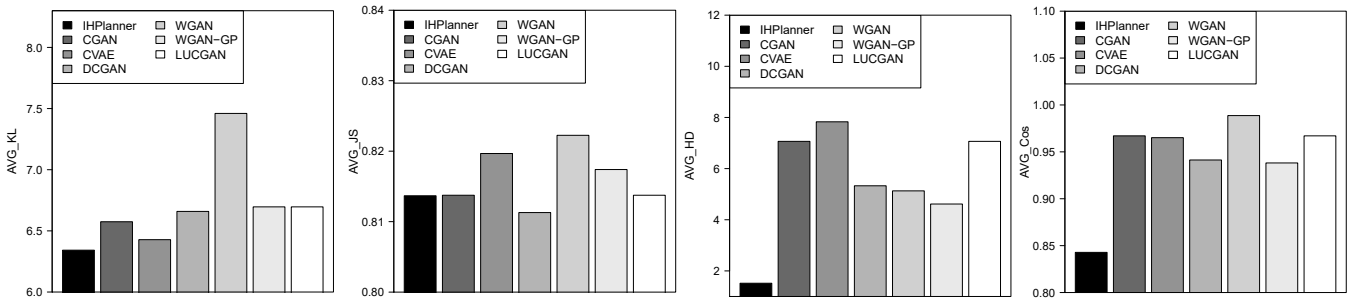
Experiments

Experimental Setup

Data Description. Our research focuses on Beijing. The data collection process is as follows: we first crawled 2990 residential communities from soufun.com and downloaded 328,668 POIs with 20 distinct POI categories from openstreetmap.org to construct land-use configuration samples referring to (Wang et al. 2020). Then, we collected taxi trajectories from the T-drive project (Yuan et al. 2010) and downloaded road networks and POIs from openstreetmap.org. to discover urban functional zones referring to (Yuan et al. 2014). Next, we used housing price data crawled from soufun.com, mobile check-ins crawled from weibo.com, taxi trajectories, and POIs to extract socioeconomic features of geospatial contexts. Moreover, we utilized the green rate including in crawled residential community data to construct human instructions.

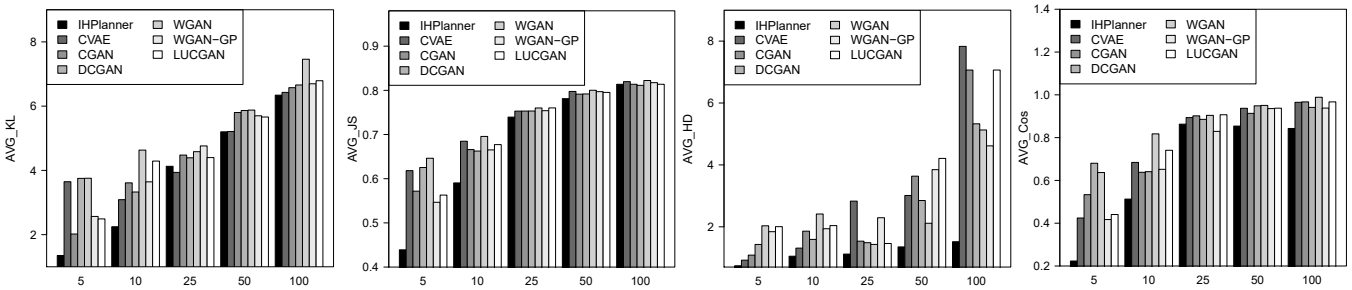
Evaluation Metrics There are five human instructions (i.e., green rate level) in our dataset: Green0, Green1, Green2, Green3, Green4. From left to right, the green rate of the land-use configuration increases. To assess the generation performance quantitatively, we adopted distribution distances as the evaluation metrics. The reason is that the data distribution of land-use configurations can be divided into different parts according to human instructions. Our planner generates a land-use configuration based on a specific human instruction. Thus, the generated configuration should be close to its green rate level's data distribution part and far from other parts. Motivated by this idea, we used four evaluation metrics: 1) **Average Kullback-Leibler (KL) Divergence (Kullback and Leibler 1951):** AVG_KL $= \frac{\sum_{j=1}^{5} w_j \cdot KL(P_j, \widehat{P}_j)}{\sum_{j=1}^{5} w_j}$. 2) **Average Jensen-Shannon (JS) Divergence (Endres and Schindelin 2003):** AVG_JS $= \frac{\sum_{j=1}^{5} w_j \cdot JS(P_j, \widehat{P}_j)}{\sum_{j=1}^{5} w_j}$. 3) **Average Hellinger Distance (HD) (Hellinger 1909):** AVG_HD $= \frac{\sum_{j=1}^{5} w_j \cdot HD(P_j, \widehat{P}_j)}{\sum_{j=1}^{5} w_j}$. 4) **Average Cosine Distance (Cos) (Singhal et al. 2001):** AVG_Cos $= \frac{\sum_{j=1}^{5} w_j \cdot Cos(P_j, \widehat{P}_j)}{\sum_{j=1}^{5} w_j}$. In all metric equations, j denotes the human instruction, w_j is the number of land-use configurations belonging to j; P_j denotes the distribution of original configurations of j; $\widehat{P}_j$ indicates the distribution of generated configurations of j; For all four metrics, the lower the metric value is, the better the generation performance is.

Baseline Models IHPlanner was compared with the following baseline models: **LUCGAN:** (Wang et al. 2020) can generate an urban plan for a geographical area according to the socioeconomic features of geospatial contexts. **CGAN:** (Mirza and Osindero 2014) can create ideal data samples (*e.g.* image, text, speech) based on conditional inputs. **CVAE:** (Sohn, Lee, and Yan 2015) is similar to CGAN yet replacing the generative model with variational autoencoder. **DCGAN:** (Radford, Metz, and Chintala 2015) is a classical image generation framework, and it has been adopted into spatiotemporal domain to capture geospatial patterns. **WGAN:** (Arjovsky, Chintala, and Bottou 2017) is an enhanced GAN, which overcomes the instability of the classical GAN and accelerates it. **WGAN-GP:** (Gulrajani et al. 2017) is an enhanced WGAN, which uses gradient penalty to replace weights clipping for improving stability. Besides, we developed four model variants to conduct ablation studies: i) **IHPlanner$^-$** removes the conditioning



(a) AVG_KL (b) AVG_JS (c) AVG_HD (d) AVG_Cos

Figure 6: Overall Performance in terms of all evaluation metrics.



(a) AVG_KL (b) AVG_JS (c) AVG_HD (d) AVG_Cos

Figure 7: The influence of different settings of N for land-use configuration generation.

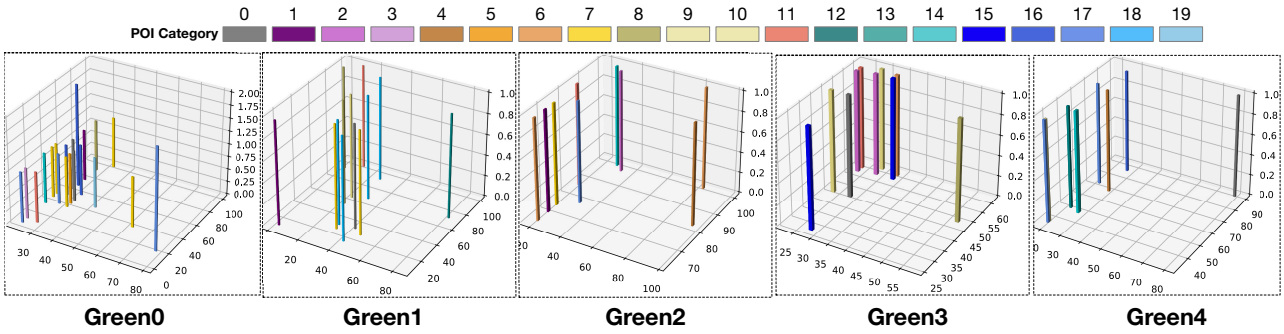
augmentation module; ii) **IHPlanner*** removes the multi-head attention module; iii) **IHPlanner**$^{'}$ removes the input of human instruction; iv) **IHPlanner**$^+$ removes the input of geospatial contexts. We randomly split the dataset into two independent sets. The prior 90% is the train set, and the remaining 10% is the test set. We provided other experimental details in the technical appendix.

Experimental Results

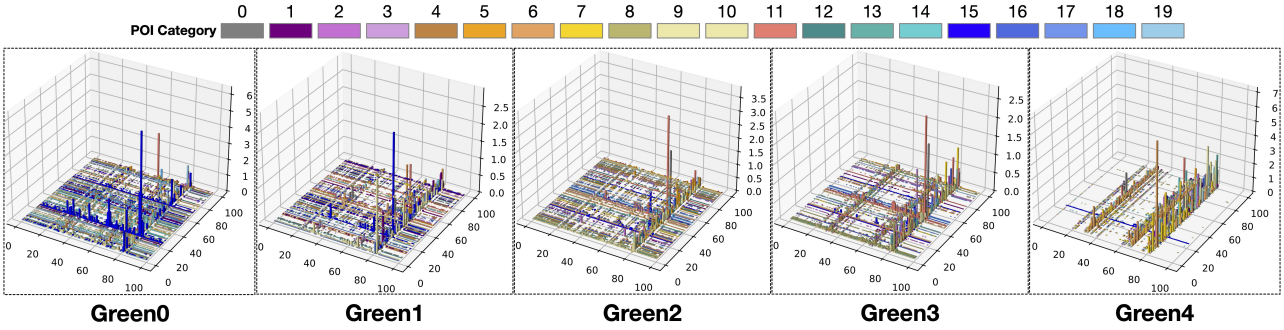
Overall Comparison This experiment aims to answer: *Can our method (IHPlanner) effectively generate land-use configurations considering human instructions and geospatial contexts?* Figure 6 shows the overall comparison results in terms of all evaluation metrics. We observed that IHPlanner outperforms all baseline models. There are two underlying drivers: i) the functionalizer module effectively projects the planning semantics of human instructions and geospatial contexts into urban functionality projections. Such projections help IHPlanner to understand the planning semantics further and generate human-friendly and environment-friendly urban plans. ii) By taking into account hierarchical zone-grid and hierarchical zone-zone dependencies in planning, IHPlanner gains suitable generation constraints for developing desirable land-use configurations.

Robustness Check This experiment aims to answer: *Is IHPlanner robust and stable when confronted with different-scale land-use configuration generation tasks?* We validated the robustness of IHPlanner by changing the value of N that is used to partition the geographical area from 5, to 10, to 25, to 50, to 100, respectively. The greater the value of N is, the finer the land-use configuration is. Figure 7 shows the comparison results in terms of all evaluation metrics. We noticed that IHPlanner outperforms all baseline models regardless of the value of N. This observation indicates that IHPlanner is more effective in perceiving the requirements of urban planning, resulting from the urban projection process of the functionalizer. Thus, our method can keep excellent and robust generation performance. Moreover, we observed that as the value of N increases, the generation performance of IHPlanner downgrades relatively. A potential reason is that the fine-grained land-use generation task requires capturing more planning details, which raises planning difficulties and thus results in poorer performance.

Visualization Analysis of the Generated Land-use Configurations. Figure 8 illustrates the visualizations of original and generated land-use configurations. In each subfigure, the left color legend provides the mapping correlation between POI categories and colors; the right 3D space exhibits the POI distribution of a land-use configuration sample; the height of each color bar shows the POIs number at the corresponding location; the text label under each subfigure is the associated human instruction. We found that the generated configurations are more organized and capture more planning details than original ones. In the meantime, we observed that as the green rate level increases, business-related POIs (e.g., POI category 15, 16) decrease, and tourism-related POIs (e.g., POI category 7, 10) increase.



(a) Original land-use configuration under different human instructions



(b) Generated land-use configuration under different human instructions

Figure 8: Visualization comparison between original land-use configurations and generated land-use configurations

Thus, this observation indicates that IHPlanner can perceive human instructions and surrounding contexts to produce excellent urban plans providing insights to urban experts.

Related Works

Deep Generative Learning. There are three kinds of popular approaches in the deep generative learning domain: normalizing flows (NF), variational autoencoders (VAE), and generative adversarial networks (GAN). NF refers to a set of generative models with tractable distributions where both sampling and density evaluation can be efficient and exact (Kobyzev, Prince, and Brubaker 2020). VAE is capable of learning the latent representations of data and providing deep inference models (Kipf and Welling 2016). GAN is able to simulate the distribution of real data by the competing of generator and discriminator under a zero-sum game setting (Creswell et al. 2018).

Attention Models. Attention mechanism gradually becomes a necessary technical module in novel deep neural networks for improving model performance (Wu et al. 2020). For instance, (Lee et al. 2018) presented a stacked cross attention framework to discover the latent alignments between the image space and the text space for conducting more accurately image-text matching. Wang *et al.* provided a knowledge graph (KG) attention network that captures the high-order connectivity of KG to improve the recommendation performance (Wang et al. 2019).

Urban Planning. With the popularity of the concept of smart city, urban planning plays a more important role in the urban development (Wang et al. 2018, 2021c). For instance, (Khansari, Mostashari, and Mansouri 2014) studied the impact of the smart city on urban sustainability and urban planning. Recently, the remarkable success of deep learning has led researchers to think about how to utilize artificial intelligence to improve the efficiency of urban planning (Shen et al. 2020). For example, (Shen et al. 2020) utilized a GAN model to fill the urban elements in road map figures to produce the final urban plan. Compared with these works, IHPlanner is more advanced automatically and practically.

Conclusion Remarks

In this paper, we propose a revolutionary deep urban planner, namely IHPlanner. To develop practical planning solutions based on planning requirements, we automate the urban planning process using a hierarchical generation method inspired by the workflow of urban experts. The input of IHPlanner is the integrated embedding of human instructions and geospatial contexts, which makes IHPlanner able to produce desirable urban plans according to human intention. The Functionalizer is a significant innovation in IHPlanner, which perceives the planning dependencies between different urban zones via the multi-attention mechanism. Extensive experiments and case studies demonstrate the effectiveness and superiority of IHPlanner. In the future, we plan to add more human-machine interactions to make the automated urban planner become more practical.

Acknowledgments

This research was partially supported by the National Science Foundation (NSF) via the grant numbers: 2040950, 2006889, 2045567.

References

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein Generative Adversarial Networks. In Precup, D.; and Teh, Y. W., eds., *Proceedings of the 34th International Conference on Machine Learning*, volume 70, 214–223. International Convention Centre, Sydney, Australia: PMLR.

Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; and Bharath, A. A. 2018. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1): 53–65.

Endres, D.; and Schindelin, J. 2003. A new metric for probability distributions. *IEEE Transactions on Information Theory*, 49(7): 1858–1860.

Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; and Courville, A. 2017. Improved training of wasserstein GANs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 5769–5779.

Hellinger, E. 1909. Neue Begründung der Theorie quadratischer Formen von unendlichvielen Veränderlichen. *Journal für die reine und angewandte Mathematik*, 1909(136): 210–271.

Khansari, N.; Mostashari, A.; and Mansouri, M. 2014. Impacting sustainable behavior and planning in smart city. *International journal of sustainable land Use and Urban planning*, 1(2).

Kipf, T. N.; and Welling, M. 2016. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*.

Kobyzev, I.; Prince, S.; and Brubaker, M. 2020. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Kullback, S.; and Leibler, R. A. 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1): 79–86.

Larsen, A. B. L.; Sønderby, S. K.; Larochelle, H.; and Winther, O. 2016. Autoencoding beyond pixels using a learned similarity metric. In *ICML*.

Lee, K.-H.; Chen, X.; Hua, G.; Hu, H.; and He, X. 2018. Stacked Cross Attention for Image-Text Matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Mirza, M.; and Osindero, S. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*.

Radford, A.; Metz, L.; and Chintala, S. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

Shen, J.; Liu, C.; Ren, Y.; and Zheng, H. 2020. Machine Learning Assisted Urban Filling. In *Proceddings of the 25th CAADRIA Conference*. Bangkok, Thailand: CUMINCAD.

Singhal, A.; et al. 2001. Modern information retrieval: A brief overview. *IEEE Data Eng. Bull.*, 24(4): 35–43.

Sohn, K.; Lee, H.; and Yan, X. 2015. Learning Structured Output Representation using Deep Conditional Generative Models. *Advances in Neural Information Processing Systems*, 28: 3483–3491.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Wang, D.; Fu, Y.; Liu, K.; Chen, F.; Wang, P.; and Lu, C.-T. 2021a. Automated Urban Planning for Reimagining City Configuration via Adversarial Learning: Quantification, Generation, and Evaluation. *ACM Transactions on Spatial Systems and Algorithms*.

Wang, D.; Fu, Y.; Wang, P.; Huang, B.; and Lu, C.-T. 2020. Reimagining City Configuration: Automated Urban Planning via Adversarial Learning. In *Proceedings of the 28th International Conference on Advances in Geographic Information Systems*, 497–506.

Wang, D.; Liu, K.; Johnson, P.; Sun, L.; Du, B.; and Fu, Y. 2021b. Deep Human-guided Conditional Variational Generative Modeling for Automated Urban Planning. In *2021 IEEE International Conference on Data Mining (ICDM)*, 679–688. IEEE.

Wang, P.; Fu, Y.; Zhang, J.; Li, X.; and Lin, D. 2018. Learning urban community structures: A collective embedding perspective with periodic spatial-temporal mobility graphs. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 9(6): 1–28.

Wang, P.; Liu, K.; Wang, D.; and Fu, Y. 2021c. Measuring Urban Vibrancy of Residential Communities Using Big Crowdsourced Geotagged Data. *Frontiers in big Data*, 4.

Wang, X.; He, X.; Cao, Y.; Liu, M.; and Chua, T.-S. 2019. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 950–958.

Wu, B.; Xu, C.; Dai, X.; Wan, A.; Zhang, P.; Yan, Z.; Tomizuka, M.; Gonzalez, J.; Keutzer, K.; and Vajda, P. 2020. Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*.

Wu, L.; Chen, Y.; Shen, K.; Guo, X.; Gao, H.; Li, S.; Pei, J.; and Long, B. 2021. Graph neural networks for natural language processing: A survey. *arXiv preprint arXiv:2106.06090*.

Wu, L.; Cui, P.; Pei, J.; Zhao, L.; and Song, L. 2022. Graph neural networks. In *Graph Neural Networks: Foundations, Frontiers, and Applications*, 27–37. Springer, Singapore.

Ye, X.; Du, J.; and Ye, Y. 2021. MasterplanGAN: Facilitating the smart rendering of urban master plans via generative adversarial networks. *Environment and Planning B: Urban Analytics and City Science*, 23998083211023516.

Yuan, J.; Zheng, Y.; Zhang, C.; Xie, W.; Xie, X.; Sun, G.; and Huang, Y. 2010. T-Drive: Driving Directions Based on Taxi Trajectories. In *Proceedings of 18th ACM SIGSPATIAL Conference on Advances in Geographical Information Systems*.

Yuan, N. J.; Zheng, Y.; Xie, X.; Wang, Y.; Zheng, K.; and Xiong, H. 2014. Discovering urban functional zones using latent activity trajectories. *IEEE Transactions on Knowledge and Data Engineering*, 27(3): 712–725.