A Discussions

In this paper, we present PointGPT, an innovative extension of the GPT framework to point clouds. Our scalable approach allows for learning high-capacity models that generalize well, achieving state-of-the-art performance on various downstream tasks. However, there are still some aspects that need to be improved. (I) Although dense self-attention is deliberately chosen for its domain-agnostic nature and widespread utilization in various domains, its quadratic computational complexity poses memory and computational challenges when pre-training or fine-tuning PointGPT on large-scale point clouds. (II) The hybrid datasets collected for PointGPT are significantly smaller in scale compared to those in NLP (2; 5) and image processing (15; 8) domains, thereby imposing limitations on the training of larger models. In the future, we aim to further explore the performance boundaries of PointGPT and extend its application to a wider range of downstream tasks.

B Model Architecture

Following previous studies (14; 10), our original PointGPT model employs the ViT-S configuration (15) for the extractor module, enabling the intuitive comparison with these prior methods. The model architecture details of PointGPT-S can be found in Table 1(a). Building upon the PointGPT-S model, two higher-capacity models, PointGPT-B and PointGPT-L, are introduced by scaling the extractor to the ViT-B and ViT-L configurations, respectively. The detailed model architectures of PointGPT-B and PointGPT-L are presented in Table 1(b) and Table 1(c).

C Datasets

C.1 Unlabeled Hybrid Dataset

Our unlabeled hybrid dataset contains approximately 300K point clouds in total, with each point cloud containing 1024 points. The unlabeled hybrid dataset is constructed by aggregating point clouds from seven existing point cloud datasets. These datasets can be classified into three categories based on their sources: clean object datasets that acquire point clouds from clean 3D models, comprising ModelNet40 (13), PartNet (9), and ShapeNet (3); indoor datasets that extract point clouds from real-world indoor scans, including S3DIS (1), ScanObjectNN (12), and SUN RGB-D (11); and outdoor dataset Semantic3D (6) that extracts point clouds from real-world outdoor scans. For each dataset, we only utilize the point clouds in the training split for model pre-training.

ModelNet40: The ModelNet40 dataset consists of 12,311 meshed computer-aided design (CAD) models, spanning 40 categories, of which 80% are designated for training and the remaining are designated for testing. For integration into the hybrid datasets, we uniformly sample 1024 points from each model in the ModelNet40 dataset.

PartNet: The PartNet dataset collects 573,585 part instances across 26,671 3D models, covering 24 distinct object categories. Each 3D object within the PartNet dataset is annotated with fine-grained, instance-level, and hierarchical 3D part information. As part of our data collection process, these 3D models are downsampled into 1024 points and incorporated into our hybrid datasets.

ShapeNet: The ShapeNet dataset encompasses a wide range of 55 common object categories, comprising more than 50,000 distinct 3D models. Following Point-MAE (10), we partition the ShapeNet dataset into separate training and validation sets. Subsequently, we downsample the point clouds from the ShapeNet dataset and incorporate them into our hybrid datasets.

S3DIS: The S3DIS dataset contains five large-scale indoor areas situated within three distinct buildings. These indoor scans span a total area of 6020 square meters and encompass an extensive collection of over 215 million points with ground-truth semantic annotations. The point clouds within the dataset are further divided into 272 rooms and annotated with 13 semantic elements, with an additional label assigned for clutter. To incorporate S3DIS into the hybrid datasets and obtain labels for post-pre-training, we extract instance-level point clouds based on their corresponding semantic annotations and retain only those point clouds that exceed a minimum threshold of 1024 points.

ScanObjectNN: The ScanObjectNN dataset encompasses approximately 15,000 objects, classified into 15 distinct categories, with a total of 2902 unique object instances. To enhance the diversity of

Table 1: Architecture details of PointGPT-S, PointGPT-B, and PointGPT-L models. The extractor in PointGPT-S is configured based on the ViT-S (15), which is in line with prior methods (14; 16; 10). The extractor modules in PointGPT-B and PointGPT-L are scaled to ViT-B and ViT-L, respectively.

(a) Architecture details of PointGPT-S.

Stage	PointGPT-S	Output Size
Data	ShapeNet	1024×3
Sequencer	$\begin{matrix}\mathrm{Partition}(64, 32) \\ \mathrm{PointNet}(128, 256, 512, 384)\end{matrix}$	64×384
Extractor	$\begin{bmatrix}\mathrm{MHA}(384) \\ \mathrm{MLP}(1536)\end{bmatrix} \times 12$	64×384
Generator	$\begin{bmatrix}\mathrm{MHA}(384) \\ \mathrm{MLP}(1536)\end{bmatrix} \times 4$	64×384
Head	$\begin{matrix}\mathrm{MLP}(96) \\ \mathrm{Reshape}\end{matrix}$	$64 \times 32 \times 3$

(b) Architecture details of PointGPT-B.

Stage	PointGPT-B	Output Size
Data	UnlabeledHybrid	1024×3
Sequencer	$\begin{matrix}\mathrm{Partition}(64, 32) \\ \mathrm{PointNet}(128, 256, 512, 1024, 768)\end{matrix}$	64×768
Extractor	$\begin{bmatrix}\mathrm{MHA}(768) \\ \mathrm{MLP}(3072)\end{bmatrix} \times 12$	64×768
Generator	$\begin{bmatrix}\mathrm{MHA}(768) \\ \mathrm{MLP}(3072)\end{bmatrix} \times 4$	64×768
Head	$\begin{matrix}\mathrm{MLP}(96) \\ \mathrm{Reshape}\end{matrix}$	$64 \times 32 \times 3$

(c) Architecture details of PointGPT-L.

Stage	PointGPT-L	Output Size
Data	UnlabeledHybrid	1024×3
Sequencer	$\begin{matrix}\mathrm{Partition}(64, 32) \\ \mathrm{PointNet}(128, 256, 512, 1024, 1024)\end{matrix}$	64×1024
Extractor	$\begin{bmatrix}\mathrm{MHA}(1024) \\ \mathrm{MLP}(4096)\end{bmatrix} \times 24$	64×1024
Generator	$\begin{bmatrix}\mathrm{MHA}(1024) \\ \mathrm{MLP}(4096)\end{bmatrix} \times 4$	64×1024
Head	$\begin{matrix}\mathrm{MLP}(96) \\ \mathrm{Reshape}\end{matrix}$	$64 \times 32 \times 3$

Table 2: Components of the unlabeled hybrid dataset. The hybrid pre-training dataset is constructed by collecting point clouds from multiple diverse point cloud datasets.

Dataset	Size	Source
ModelNet40	12311	Clean 3D Models
PartNet	26671	Clean 3D Models
ShapeNet	52470	Clean 3D Models
S3DIS	8947	Indoor
ScanObjectNN	100390	Indoor
SUN RGB-D	19061	Indoor
Semantic3D	62807	Outdoor
UnlabeledHybrid	282657	Multi-source

our hybrid datasets, we incorporate three variants: OBJ_ONLY, OBJ_BG, and PB_T50_RS. The OBJ_ONLY variant exclusively comprises ground truth segmented objects, while OBJ_BG includes objects with their nearby backgrounds. Lastly, the PB_T50_RS variant introduces perturbations to simulate scenarios where bounding boxes may either over- or under-cover objects, or even result in object splitting.

SUN RGB-D: The SUN RGB-D dataset comprises a collection of 10,335 RGB-D images, captured using four distinct sensors, and annotated with both 2D and 3D bounding boxes. These RGB-D images are subsequently transformed into 3D point clouds (4). From the ground-truth 3D bounding boxes, we extract instance-level point clouds and retain those with a point count exceeding 1024. In the post-pre-training stage, the resulting point clouds are annotated with semantic labels derived from the bounding boxes, encompassing 10 object categories.

Semantic3D: The Semantic3D dataset comprises a vast collection of scanned outdoor scenes, containing over 3 billion points. It consists of 15 scenes designated for training and an additional 15 scenes for testing. To partition the outdoor scans, we employ a voxelization process, utilizing voxel dimensions of $[0.8 \times 0.8 \times 0.8]$. Following this step, only the resulting point clouds with a point count exceeding 1024 are retained.

C.2 Labeled Hybrid Dataset

The labeled hybrid dataset is constructed based on the unlabeled hybrid dataset, excluding the Semantic3D dataset due to the absence of semantic annotations for the generated point clouds, obtaining a total of 219,850 point clouds. The semantic labels are obtained by aligning the semantic labels of the collected point clouds, covering 87 categories.

D Implementation Details

Our PointGPT models are pre-trained and post-pre-trained with a batch size of 128 for 300 epochs. We employ the AdamW optimizer (7) with an initial learning rate of $1e-3$ and a weight decay of 0.05. The learning rate is linearly scaled based on the total batch size using the formula $lr = base\_lr \times batch\_size/256$. Additionally, we apply the cosine learning rate schedule to gradually decay the learning rate over epochs. The specific settings for pre-training and post-pre-training are presented in Table 3. Remarkably, we successfully train the PointGPT-L model using merely two Nvidia GeForce RTX 3090 graphics cards.

E Ablation Studies on Post-pre-training

The effect of the post-pre-training stage is analyzed in Table 4, the post-pre-training stage significantly improves the classification accuracy of PointGPT models on the real-world ScanObjectNN dataset, but marginally enhances the classification accuracy on the ModelNet40 dataset and the mIoU on the ShapeNetPart dataset. Further analyses for each dataset are presented below:

Table 3: Implementation details for the pre-training and post-pre-training stages.

(a) Pre-training setting.

Config	Value
optimizer	AdamW
base learning rate	0.002
weight decay	0.05
batch size	128
learning rate schedule	cosine decay
warmup epoch	10
epoch	300
augmentation	scale and translate
mask ratio	0.7

(b) Post-pre-training setting.

Config	Value
optimizer	AdamW
base learning rate	0.002
weight decay	0.05
batch size	128
learning rate schedule	cosine decay
warmup epoch	10
epoch	300
dropout	0.5
drop path	0.3

Table 4: Ablation studies on post-pre-training stage. We report the fine-tuned classification accuracy on ScanObjectNN, ModelNet40 datasets, and the mIoU across all classes (Cls.) and all instances (Inst.) on the ShapeNetPart dataset. All results are expressed as percentages.

Methods	ScanObjectNN			ModelNet40		ShapeNetPart	
	OBJ_BG	OBJ_ONLY	PB_T50_RS	1k P	8k P	Cls.mIoU	Inst.mIoU
without post-pre-training							
PointGPT-B	93.6	92.5	89.6	94.2	94.4	84.5	86.4
PointGPT-L	95.7	94.1	91.1	94.5	94.7	84.7	86.5
with post-pre-training							
PointGPT-B	95.8 (+2.2)	95.2 (+2.7)	91.9 (+2.3)	94.4 (+0.2)	94.6 (+0.2)	84.5 (+0.0)	86.5 (+0.1)
PointGPT-L	97.2 (+1.5)	96.6 (+2.5)	93.4 (+2.3)	94.7 (+0.2)	94.9 (+0.2)	84.8 (+0.1)	86.6 (+0.1)

ScanObjectNN: The partial visibility of objects in the ScanObjectNN dataset, particularly when permutation is introduced, poses a significant challenge. We find that this challenge can be effectively mitigated by performing post-pre-training on the unlabeled hybrid dataset. The unlabeled hybrid dataset aligns the semantic labels across different datasets, which allows the clean and complete 3D object models to provide valuable guidance for learning on the ScanObjectNN dataset.

ModelNet40: The ModelNet40 dataset exhibits a long-tail distribution, which is further emphasized when it is included in the unlabeled hybrid dataset. Only approximately half of the categories in the ModelNet40 dataset are shared with other datasets. Therefore, the post-pre-training of the model does not lead to significant improvements in classification accuracy on the ModelNet40 dataset.

ShapeNetPart: The main reason we analyze that limits the performance improvement of post-pre-training on part segmentation tasks is due to (I) the limited capacity of the unlabeled hybrid dataset and (II) the inherent disparities between classification tasks and segmentation tasks, which pose obstacles in leveraging the acquired knowledge from post-pre-training to enhance the performance of the model on the ShapeNetPart dataset.

F Pre-training Results

Fig. 1 presents the qualitative results of our auto-regressive generation tasks, showcasing the effectiveness of our approach. Specifically, our method proficiently predicts subsequent point patches and accurately generates the original point cloud. The generation results verify the low information density of point clouds, such that our method is able to accurately predict patches even in cases where the point cloud is extensively masked with a high proportion. Additionally, the results suggest the feasibility of organizing point clouds in a predefined geometric order, offering a novel perspective for pre-training transformer models in point clouds.

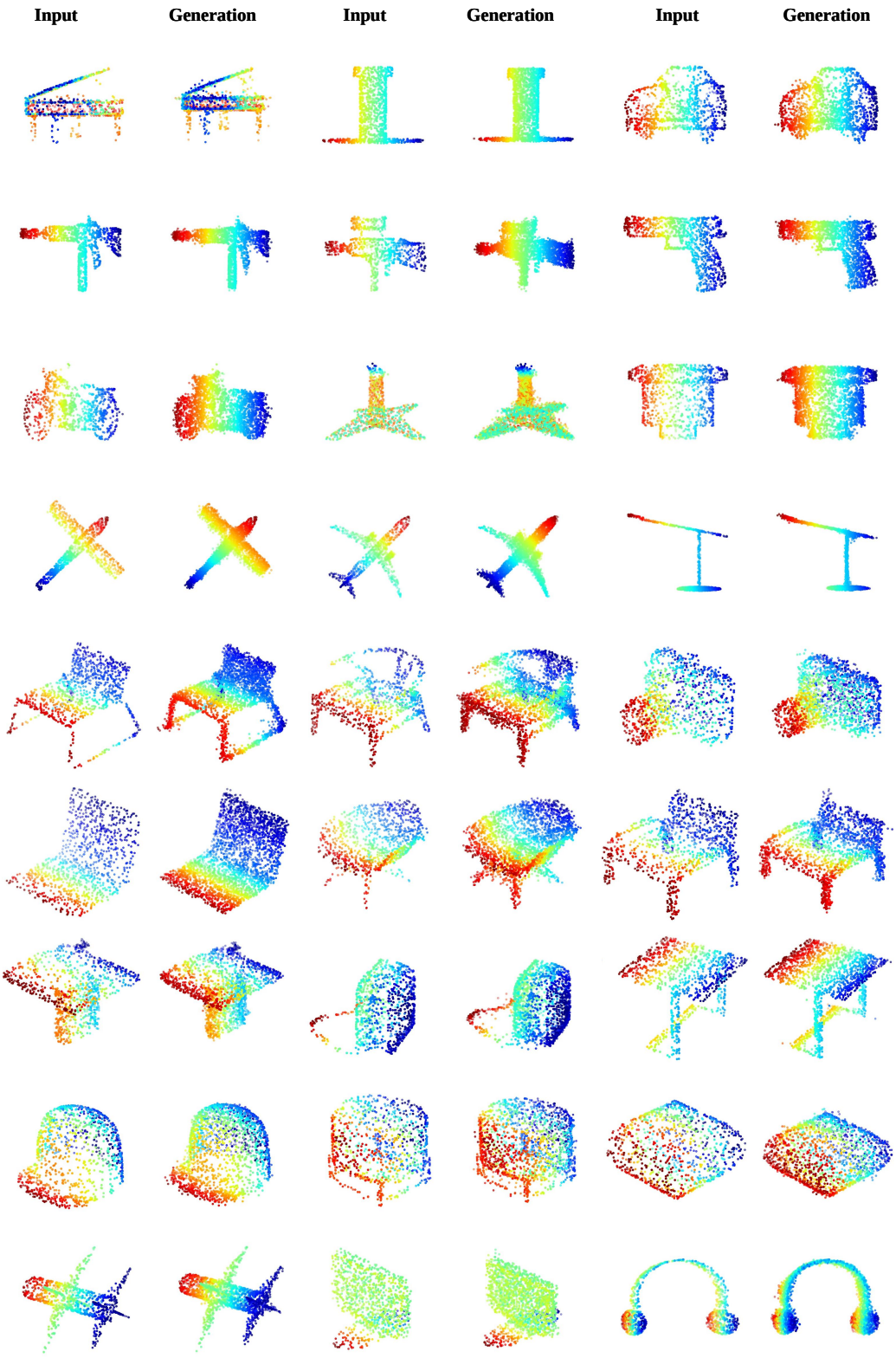

Figure 1: Generation examples on ShapeNet validation set. The original input (i.e., ground truth) and generation result of each group is shown from left to right.

References

[1] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1534–1543, 2016.

[2] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[3] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[4] MMDetection3D Contributors. MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. `https://github.com/open-mmlab/mmdetection3d`, 2020.

[5] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee, 2005.

[6] Timo Hackel, Nikolay Savinov, Lubor Ladicky, Jan D Wegner, Konrad Schindler, and Marc Pollefeys. Semantic3d. net: A new large-scale point cloud classification benchmark. *arXiv preprint arXiv:1704.03847*, 2017.

[7] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[8] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12009–12019, 2022.

[9] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918, 2019.

[10] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. *arXiv preprint arXiv:2203.06604*, 2022.

[11] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015.

[12] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Duc Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *International Conference on Computer Vision (ICCV)*, 2019.

[13] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015.

[14] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19313–19322, 2022.

[15] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12104–12113, 2022.

[16] Renrui Zhang, Ziyu Guo, Peng Gao, Rongyao Fang, Bin Zhao, Dong Wang, Yu Qiao, and Hongsheng Li. Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training. *arXiv preprint arXiv:2205.14401*, 2022.