MAE-XNT: A Foundation Model for Segmenting Neuronal Tissue Volumes Generated with X-Ray Nanotomography

Supplementary Material

7. Ablation study

We train several foundation models using different masking ratios and patch sizes following the process described in Sec. 3. Higher masking ratios and patch sizes allow for larger batch sizes, then for each training, we set the batch size to fully use the available GPU memory. We show in the first column of Tab. 7 the batch size used for each masking ratio and patch size setting. To determine the best model, we fine-tune them on the iid and ood datasets using 1%, 10% and 100% of the training data. For each model, the fine-tuning parameters have been found through a random search as described in Sec. 8. We fine-tune the models trained with a patch size of 4^3 voxel using the UNETR decoder described in Fig. 2. To fine-tune the model trained with a patch size of 8^3 voxel, we adapt the UNETR decoder to fit this patch size as described in Fig. 5. The model that performs the best on average on both the iid and ood datasets is the one trained using a masking ratio of 0.9 and a patch size of 4 (see Tab. 4). We call this model MAE-XNT.

8. Random search for cell nuclei fine-tuning

We perform a separate random search for each foundation model (trained using different masking ratios and patch sizes) to determine the best fine-tuning parameters for the cell nuclei segmentation task. To achieve that, we fine-tuned the models using 10% of the training data from the iid cell nuclei segmentation dataset (described in Sec. 4.2). We launch 30 trainings for each random search considering four parameters: the learning rate, the weight decay coefficient, the layer-wise learning rate decay and the fine-tuning time using two NVIDIA H100 GPUs. The value ranges explored in the random searches are provided in Tab. 5.

We also conduct a random search for the training from scratch considering the same hyperparameters as specified above except the layer-wise learning rate decay that is set by default to 1 (changing this parameter is irrelevant when training from scratch). The value ranges associated with the parameters considered in the random search are provided in Tab. 6.

For each trained foundation model, we provide in Tab. 7 the hyperparameters found by their associated random search. The last line of the table provides the outcome of the random search performed for trainings from scratch.

9. Random search for axon segmentation fine-tuning

We perform two additional random searches to select the hyperparameters for training on the XPRESS dataset: one for the MAE-XNT model, and one for training from scratch. To conduct this experiment, we run 30 training sessions and vary four parameters: the learning rate, the weight decay coefficient, the layer-wise learning rate decay, and the number of training steps, using two NVIDIA H200 GPUs. The value ranges for the random search on the foundation model are listed in Tab. 8, and those for the training from scratch are listed in Tab. 9. The agglomeration threshold defined in Sec. 5.2 is also optimized during this search by evaluating three candidate values $(0.5, 0.6, 0.7)$ for each training session.

The results of this random search are provided in Tab. 10 for both the foundation model and the model trained from scratch.

Table 4. Dice scores obtained for the cell nuclei segmentation task, after fine-tuning the foundation models pre-trained using different masking ratios and patch sizes.

Dataset	Masking ratio / Patch size	1%	10%	100%	Mean
iid	0.75 / 4	0.922	0.941	**0.947**	0.937
	0.80 / 4	0.933	0.941	0.946	0.940
	0.85 / 4	0.930	0.940	0.945	0.938
	0.90 / 4	**0.934**	**0.943**	**0.947**	**0.941**
	0.90 / 8	0.932	0.942	0.944	0.939
ood	0.75 / 4	0.826	0.845	0.856	0.842
	0.80 / 4	**0.831**	0.831	0.865	0.842
	0.85 / 4	0.824	0.826	**0.878**	0.843
	0.90 / 4	0.800	**0.874**	**0.878**	**0.851**
	0.90 / 8	0.761	0.814	0.834	0.803

Table 5. Value ranges of the parameters considered in the random search performed for foundation models. The random search is conducted using 10% of the iid dataset training data.

Hyperparameter	Value range
learning rate	$\{0.000005, 0.00001, 0.00005, 0.0001, 0.0005\}$
weight decay	$\{0.00001, 0.0001, 0.001\}$
training time (hours)	$\{2, 4, 6, 8\}$
layer-wise learning rate decay	$\{0.60, 0.75, 0.90, 1\}$

Table 6. Value ranges of the parameters considered in the random search performed for training from scratch. The random search is conducted using 10% of the iid dataset training data.

Hyperparameter	Value range
learning rate	$\{0.00001, 0.00005, 0.0001, 0.0005, 0.001\}$
weight decay	$\{0.00001, 0.0001, 0.001\}$
training time (hours)	$\{4, 8, 12, 16\}$

Table 7. Hyperparameters found after a random search (on the iid dataset using 10% of training data) for different pre-training settings.

Mask ratio / Patch size / Batch size	Learning rate	Weight decay	Fine-tuning time	Layer-wise learning rate decay
0.75 / 4 / 224	0.00001	0.0001	4 hours	0.75
0.80 / 4 / 256	0.000005	0.0001	6 hours	0.9
0.85 / 4 / 280	0.00001	0.001	6 hours	0.75
0.90 / 4 / 320	0.00001	0.0001	6 hours	0.75
0.90 / 8 / 1792	0.0005	0.00001	4 hours	0.75
Training from scratch	0.0001	0.00001	12 hours	-

Table 8. Value ranges of the parameters considered in the random search performed for the foundation model on the axon segmentation task.

Hyperparameter	Value range
learning rate	$\{0.002, 0.001, 0.0005, 0.00025, 0.0001, 0.00005\}$
weight decay	$\{0.01, 0.05, 0.001, 0.005\}$
training steps	$\{200, 300, 400, 500\}$
layer-wise learning rate decay	$\{0.60, 0.75, 0.90, 1\}$

Table 9. Value ranges of the parameters considered in the random search performed for training from scratch on the axon segmentation task.

Hyperparameter	**Value range**
learning rate	$\{0.005, 0.001, 0.0005, 0.0001, 0.00005, 0.00001\}$
weight decay	$\{0.01, 0.05, 0.001, 0.005, 0.0005\}$
training steps	$\{500, 1000, 1500, 2000, 3000\}$
layer-wise learning rate decay	$\{1\}$

Table 10. Hyperparameters found after the random search for the foundation model and the training from scratch on axon segmentation

	Learning rate	**Weight decay**	**Number of steps**	**Layer-wise learning rate decay**
Training from scratch	0.0001	0.05	1000	-
Foundation model	0.0005	0.05	400	0.75

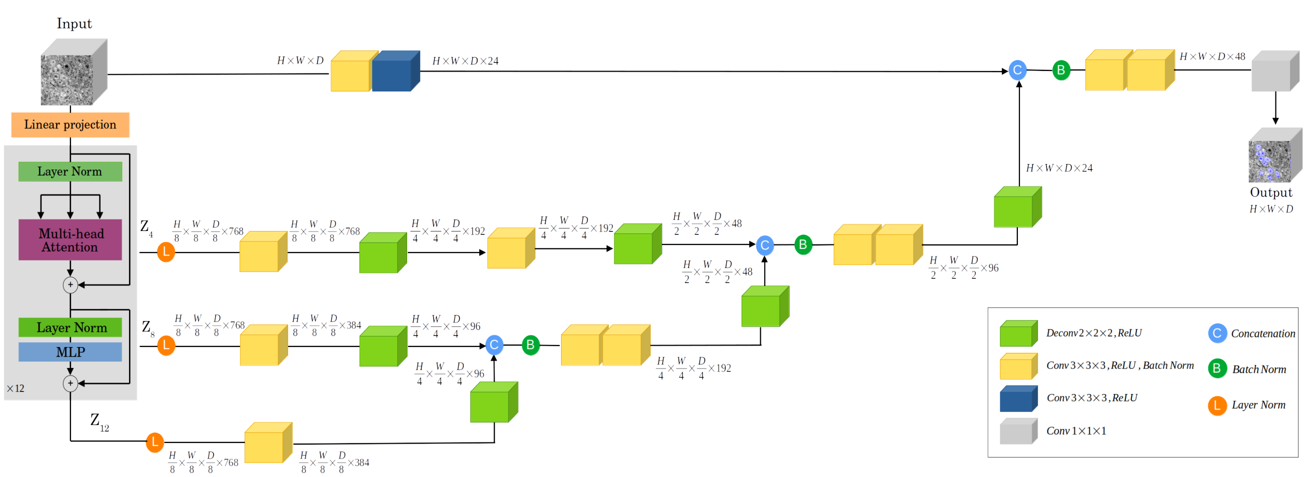

Figure 5. Adaptation of the UNETR architecture to generate patches of size 8^3.