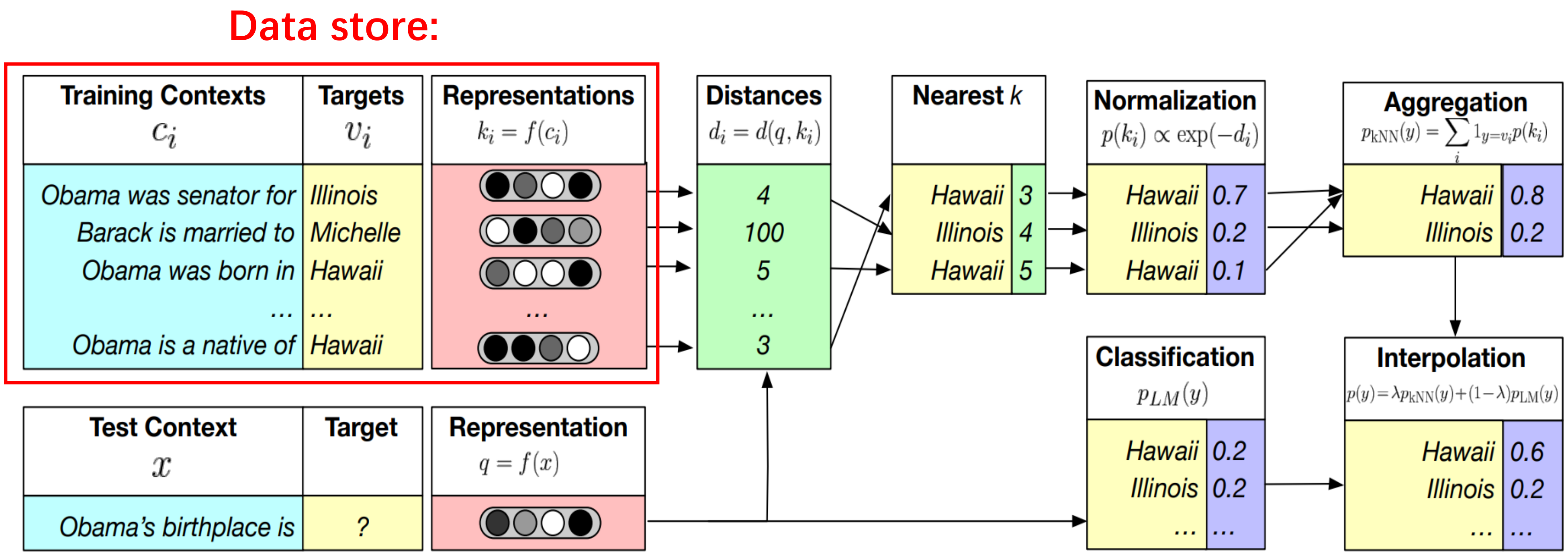
Memory augment models

- Large model store world knowledge implicitly within their parameters.

Memory augment language model

■ Overview



Data store: $(\mathcal{K}, \mathcal{V}) = \{(f(c_i), w_i) | (c_i, w_i) \in \mathcal{D}\}$ $\quad$ $p(y|x) = \lambda\, p_{\text{kNN}}(y|x) + (1 - \lambda)\, p_{\text{LM}}(y|x)$

Knn-generation: $p_{\text{kNN}}(y|x) \propto \sum_{(k_i, v_i) \in \mathcal{N}} \mathbb{1}_{y=v_i} \exp(-d(k_i, f(x)))$

[1] GENERALIZATION THROUGH MEMORIZATION:NEAREST NEIGHBOR LANGUAGE MODELS. ICLR 2020

Memory augment language model

■ Experimental results

Model	Perplexity (↓)		# Trainable Params
	Dev	Test	
Baevski & Auli (2019)	17.96	18.65	247M
+Transformer-XL (Dai et al., 2019)	-	18.30	257M
+Phrase Induction (Luo et al., 2019)	-	17.40	257M
Base LM (Baevski & Auli, 2019)	17.96	18.65	247M
+kNN-LM	**16.06**	**16.12**	247M
+Continuous Cache (Grave et al., 2017c)	17.67	18.27	247M
+kNN-LM + Continuous Cache	**15.81**	**15.79**	247M

Table 1: Performance on WIKITEXT-103. The kNN-LM substantially outperforms existing work. Gains are additive with the related but orthogonal continuous cache, allowing us to improve the base model by almost 3 perplexity points with no additional training. We report the median of three random seeds.

[1] GENERALIZATION THROUGH MEMORIZATION:NEAREST NEIGHBOR LANGUAGE MODELS. ICLR 2020

Memory augment language model

Experimental results

$$p(y|x) = \lambda\, p_{\mathrm{kNN}}(y|x) + (1-\lambda)\, p_{\mathrm{LM}}(y|x)$$

Memory Selective Size

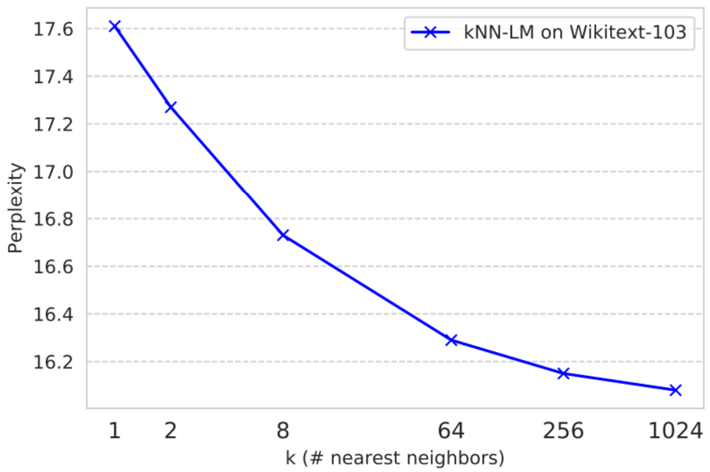


Figure 4: Effect of the number of nearest neighbors returned per word on WIKITEXT-103 (validation set). Returning more entries from the datastore monotonically improves performance.

Weight

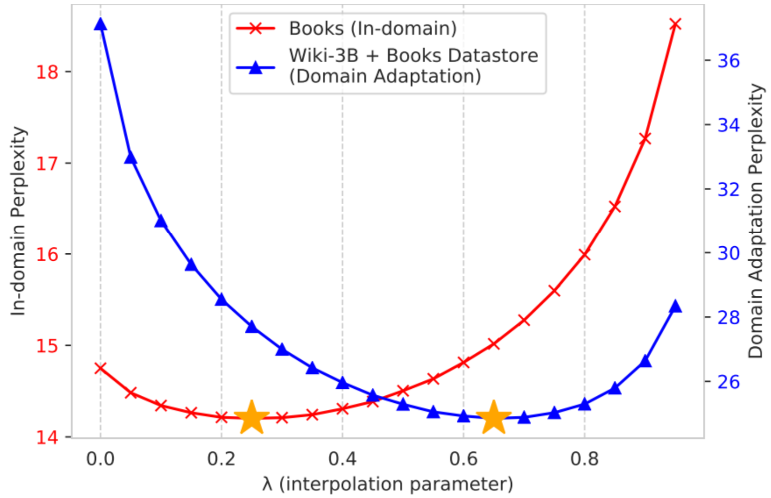


Figure 5: Effect of interpolation parameter λ on in-domain (left y-axis) and out-of-domain (right y-axis) validation set performances. More weight on p_{kNN} improves domain adaptation.

[1] GENERALIZATION THROUGH MEMORIZATION:NEAREST NEIGHBOR LANGUAGE MODELS. ICLR 2020

Memory augment language model

■ Experimental results

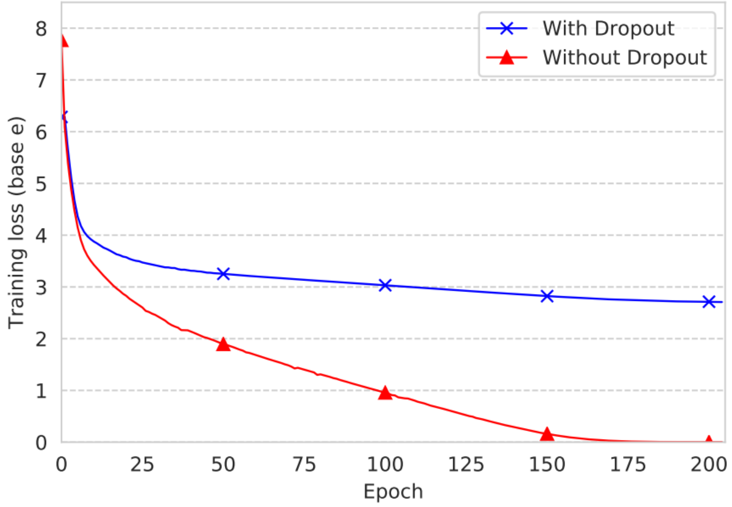
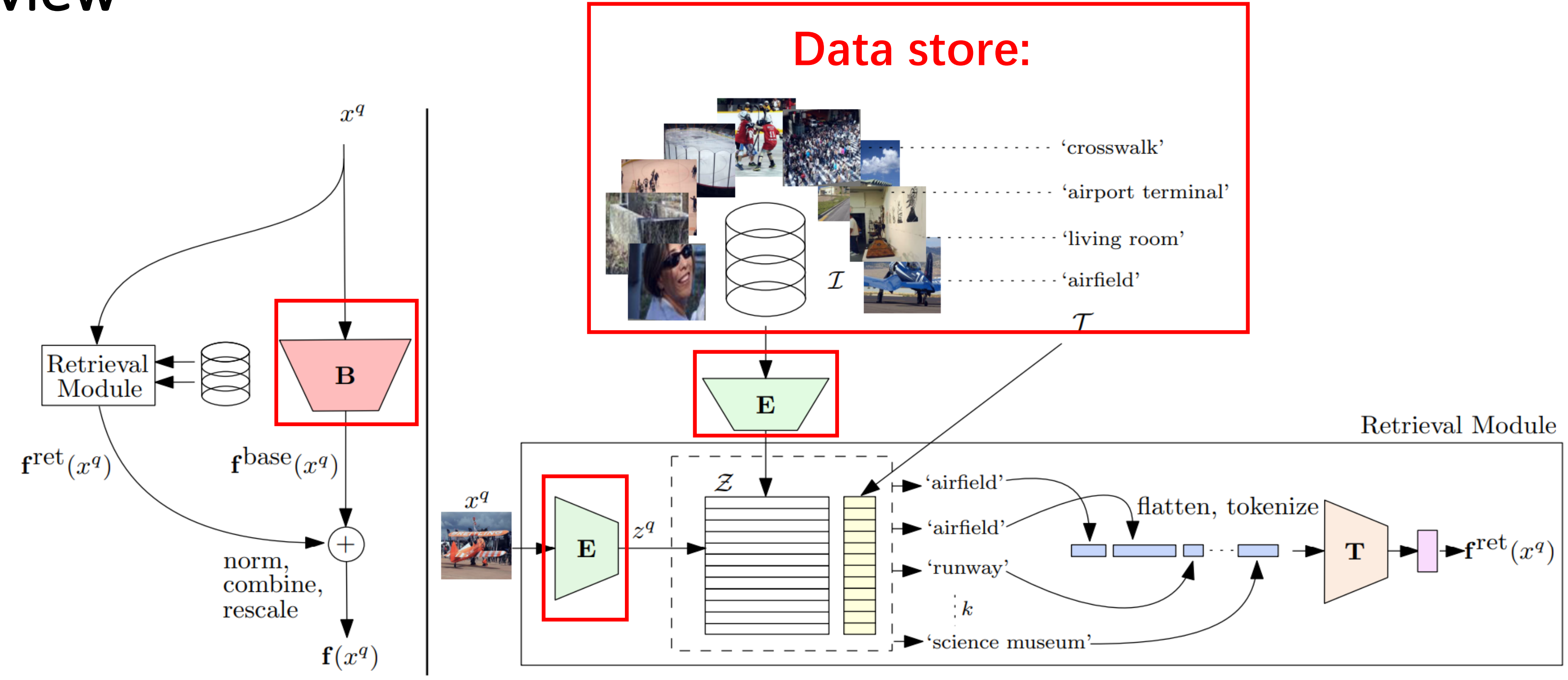


Figure 8: Training curves for the Transformer LM with and without dropout. Turning off dropout allows the training loss to go to 0, indicating that the model has sufficient capacity to memorize the training data.

[1] GENERALIZATION THROUGH MEMORIZATION:NEAREST NEIGHBOR LANGUAGE MODELS. ICLR 2020

Memory augment long tail visual recognition

■ Overview



(a) The proposed RAC architecture

(b) Retrieval module

$$\mathbf{f}(\mathbf{x}) = \frac{L}{2}\left(\frac{\mathbf{f}^{\text{ret}}(\mathbf{x})}{||\mathbf{f}^{\text{ret}}(\mathbf{x})||_2} + \frac{\mathbf{f}^{\text{base}}(\mathbf{x})}{||\mathbf{f}^{\text{base}}(\mathbf{x})||_2}\right),$$

[2] Retrieval Augmented Classification for Long-Tail Visual Recognition, CVPR 2022

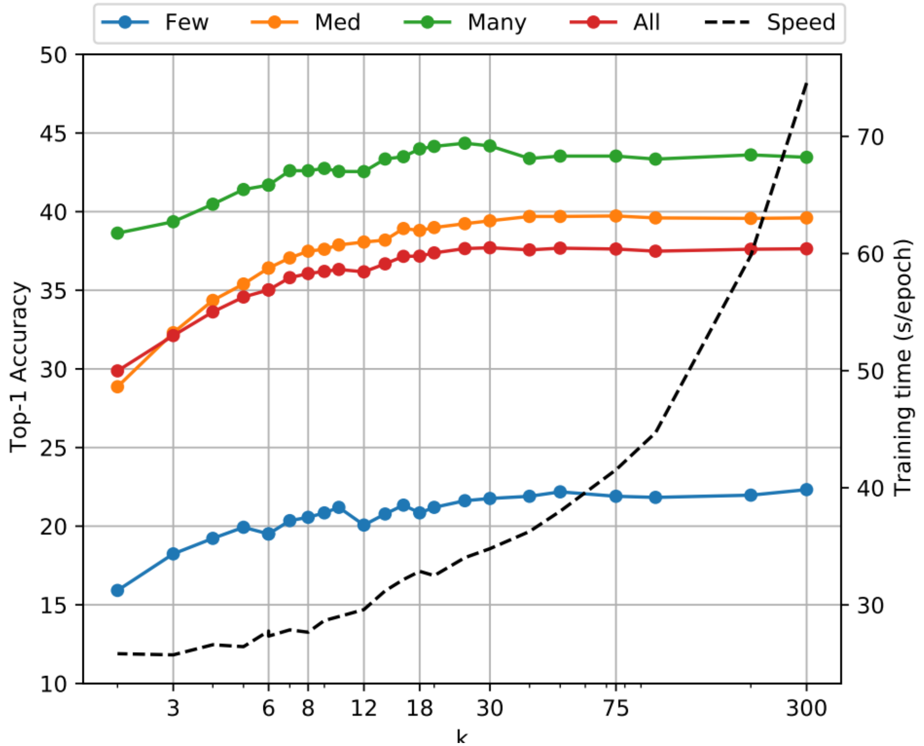
Memory augment long tail visual recognition

Experimental results

Method	Backbone	Many	Med	Few	All
Input: 224 × 224					
OLTR [33]	RN50	59	64.1	64.9	63.9
Dec. LWS [24] †	RN50	65.0	66.3	65.5	65.9
LADE [21] †	RN50	-	-	-	70.0
ALA [57] †	RN50	71.3	70.8	70.4	70.7
LACE [36]	RN50	-	-	-	71.9
RIDE [50]	RN50	70.9	72.4	73.1	72.6
TADE [56]	RN50	74.4	72.5	73.1	72.9
DisAlign [54]	RN152	-	-	-	74.1
PaCo [6]	RN152	75.0	75.5	74.7	75.2
RAC (ours)	ViT-B-16	**75.92**	**80.47**	**81.07**	**80.24**
Input: 384 × 384					
Grafit	RegNetY	-	-	-	81.2
RAC (ours)	ViT-B-16	**82.91**	**85.71**	**86.06**	**85.56**

Table 1. Historical performance on iNat under varying backbones and training schemes. †Results reproduced from [57].

Memory Selective Size

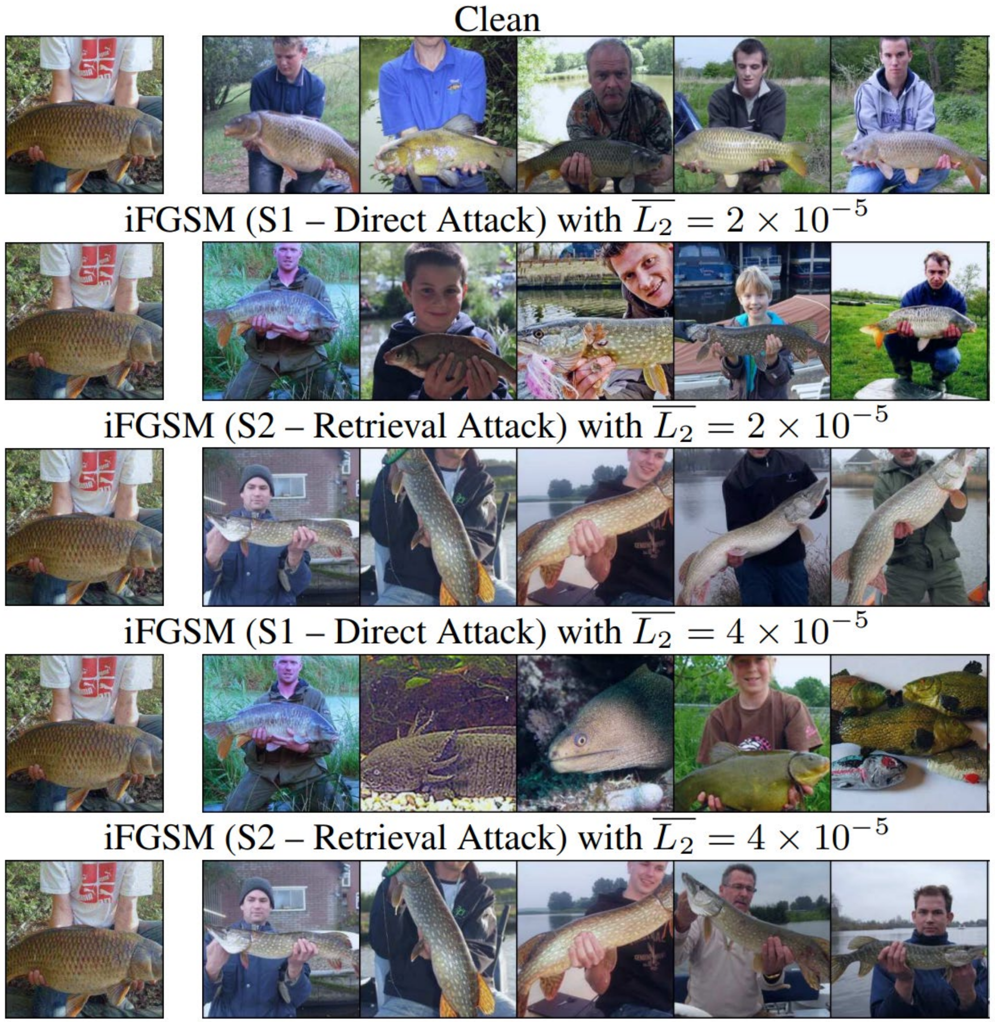


[2] Retrieval Augmented Classification for Long-Tail Visual Recognition, CVPR 2022

Memory augment CNN

■ Overview

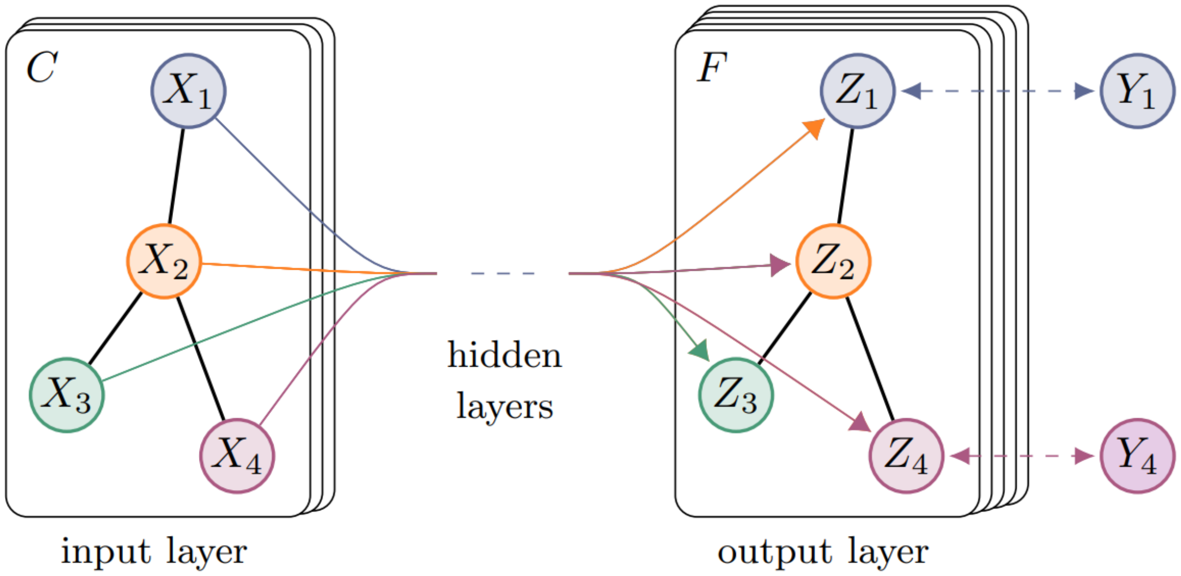
$$\mathcal{C}(F(x)) : \mathcal{P}(x) = \mathcal{P}_{\mathcal{C}(F(x))}(x) = \boxed{\sum_{k=1}^{K} \alpha_k \phi(x'_k).}$$



[2] Retrieval-Augmented Convolutional Neural Networks against Adversarial Examples, CVPR 2019

Connection with GCN

■ Overview



(a) Graph Convolutional Network

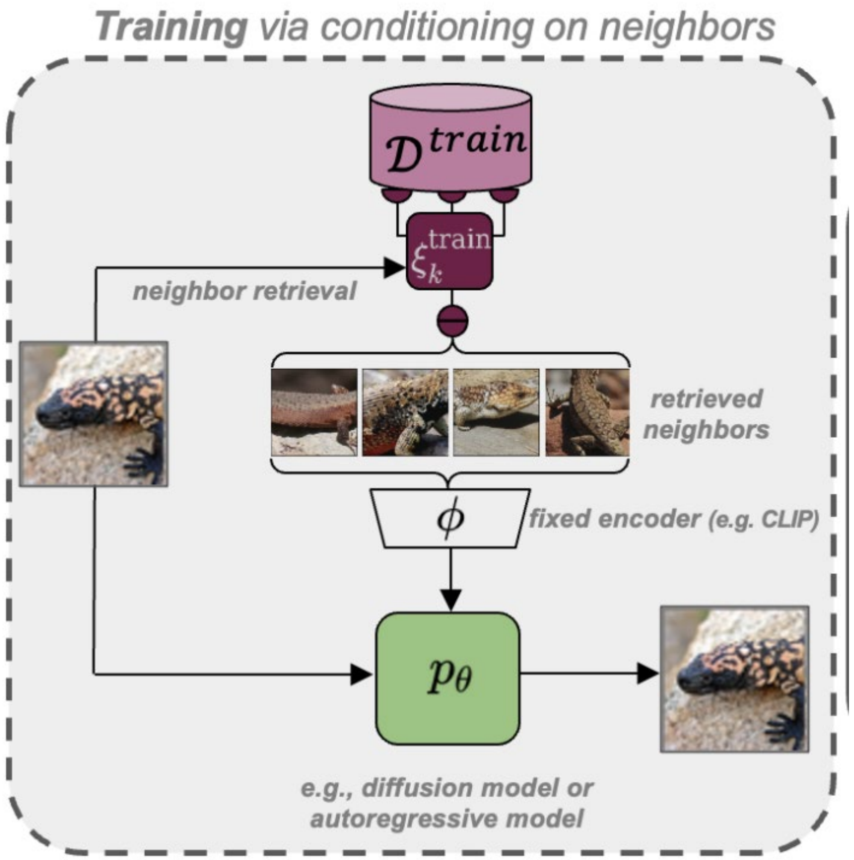
Relationship among sample

$$Z = f(X, A) = \boxed{\mathrm{softmax}\left(\hat{A}\ \mathrm{ReLU}\left(\hat{A}XW^{(0)}\right)W^{(1)}\right)\ .}$$

[3] SEMI-SUPERVISED CLASSIFICATION WITH GRAPH CONVOLUTIONAL NETWORKS. ICLR 2017

Memory augment diffusion model

■ Models



Generative model

$$p_{\theta,\mathcal{D},\xi_k}(x) = \boxed{p_\theta(x \mid \{\, \phi(y) \mid y \in \xi_k(x,\mathcal{D}) \,\})}.$$

Loss function

$$\min_\theta \mathcal{L} = \mathbb{E}_{p(x),z\sim E(x),\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t,\, t,\, \{\phi_{\text{CLIP}}(y) \mid y \in \xi_k(x,\mathcal{D})\})\|_2^2\right]$$

[2] Retrieval-Augmented Diffusion Models, Neurips2022

Memory augment diffusion model

■ Experiment results

Memory Selective Size

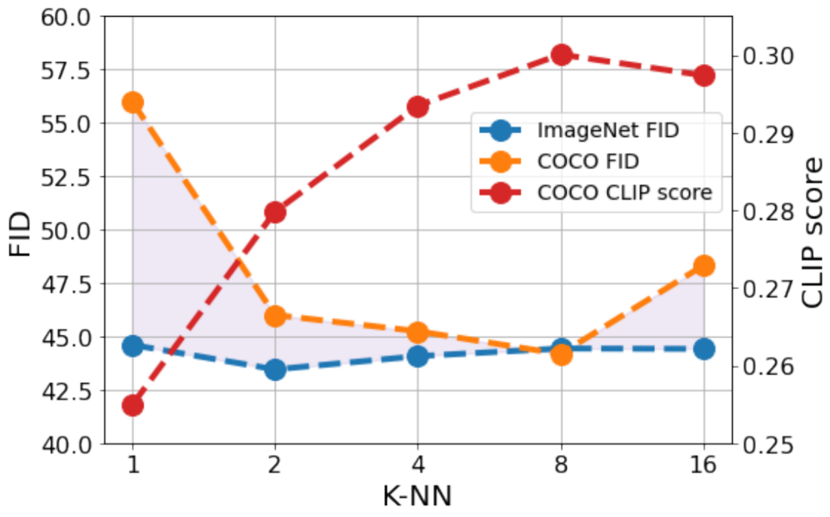
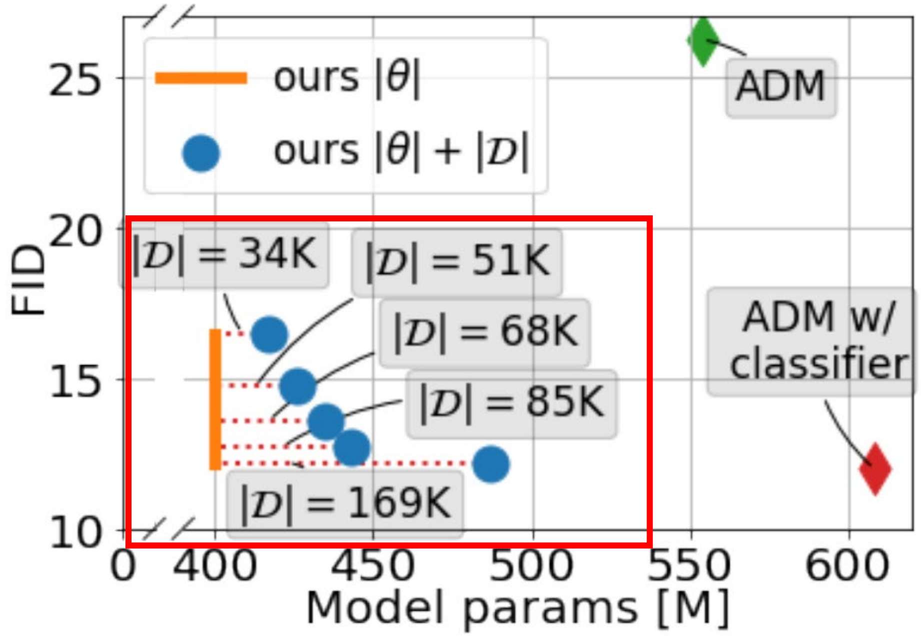


Figure 8: We observe that the number of neighbors k_{train} retrieved during training significantly impacts the generalization abilities of *RDM* . See Sec. 4.2.



[2] Retrieval-Augmented Diffusion Models, Neurips2022

Memory augment diffusion model

■ Models

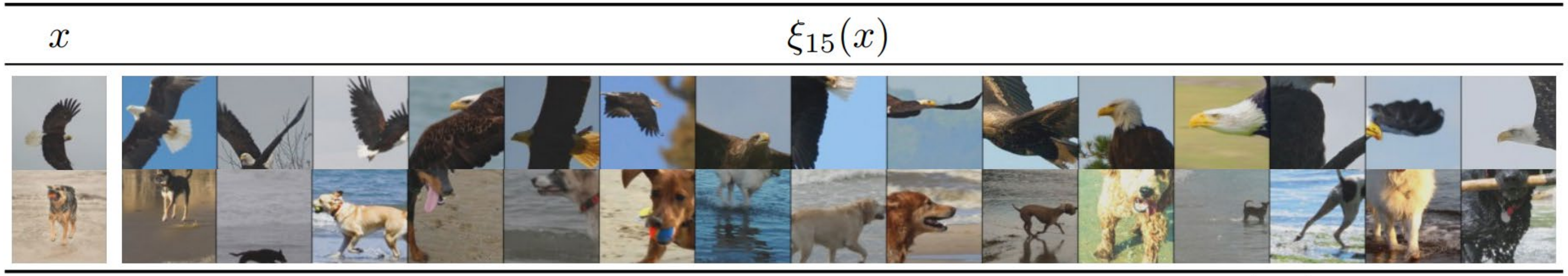


Figure 4: $k = 15$ nearest neighbors from $\mathcal{D}$ for a given query x when parameterizing $d(x, \cdot)$ with CLIP [57].