Theory and Experiments on Vector Quantized Autoencoders

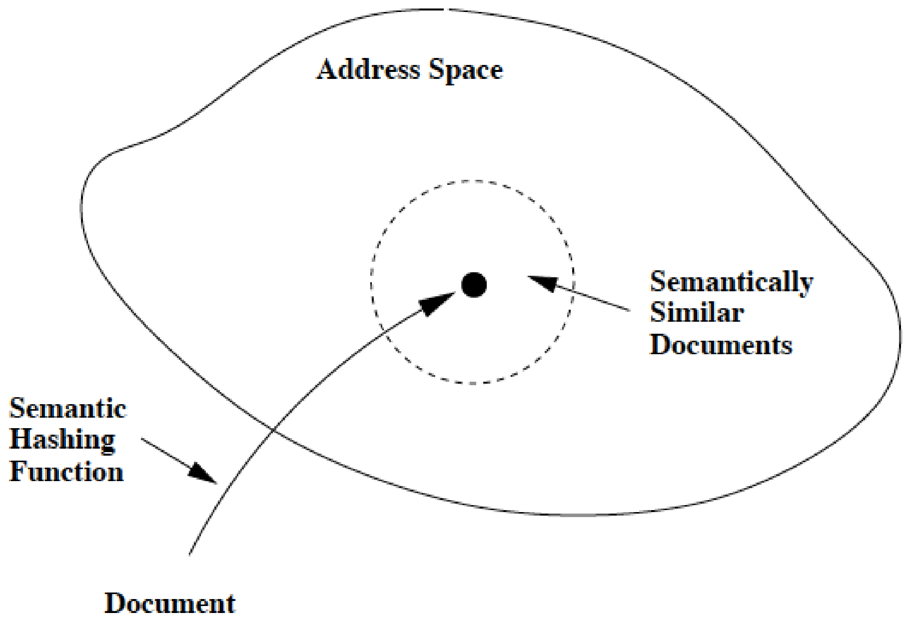
Van den Oord et al.

Park Jungsoo

Data Mining & Information Systems Lab.
Department of Computer Science and Engineering,
College of Informatics, Korea University

1 Introduction

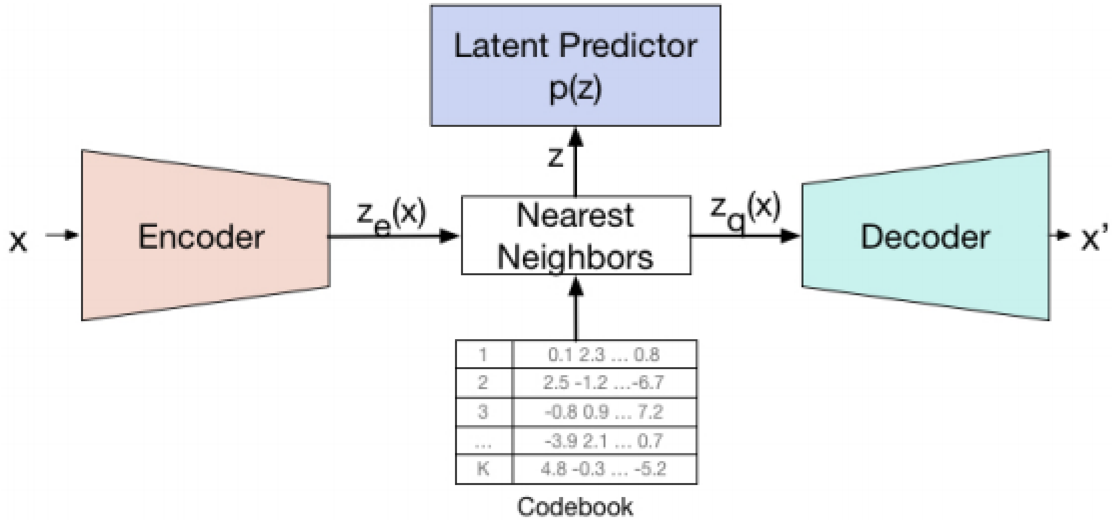
Why Discrete Latent Representation?



- Computational Efficiency
- Interpretability and Communication
- More Natural

1 Introduction

VQ-VAE



$$z_i = \arg\min_{j \in [K]} \|z_e(x_i) - e_j\|_2$$

$$L = l_r + \beta \left\| z_e(x_i) - \mathrm{sg}\left(z_q(x_i)\right) \right\|_2 ,$$

$$z_q(x_i) = e_{z_i}$$

$$\mathrm{sg}(x) = \begin{cases} x & \text{forward pass} \\ 0 & \text{backward pass} \end{cases}$$

1 Introduction

EMA Update ver.

$$c_j \leftarrow \lambda c_j + (1-\lambda) \sum_i \mathbb{1}\left[z_q(x_i) = e_j\right],$$

$$e_j \leftarrow \lambda e_j + (1-\lambda) \sum_i \frac{\mathbb{1}\left[z_q(x_i) = e_j\right] z_e(x_i)}{c_j},$$

- Calculation of averages of different subsets of the full data set.
- When used in updating embedding vectors, (instead of gradient) more stable in training.

EM Algorithm

1. **E step:** $(z_1, \ldots, z_N) \leftarrow \arg\max_{z_1,\ldots,z_N} P_\Theta(x_1, \ldots, x_N, z_1, \ldots, z_N)$,
2. **M step:** $\Theta \leftarrow \arg\max_\Theta P_\Theta(x_1, \ldots, x_N, z_1, \ldots, z_N)$

K-Means Clustering is one of EM-Algorithm

$$\Theta = \langle \mu^1, \ldots, \mu^K \rangle, \quad \mu^k \in R^D.$$

1. **E step:** Cluster assignment is given by,

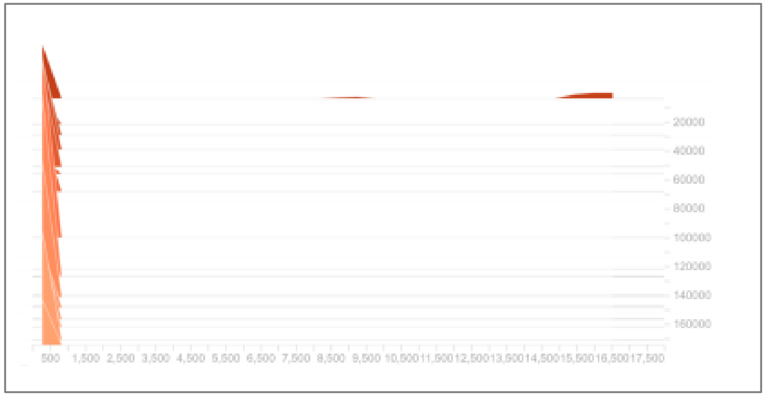
$$z_i \leftarrow \arg\min_{j \in [K]} \left\| \mu^j - x_i \right\|_2^2,$$

2. **M step:** The means of the clusters are updated as,

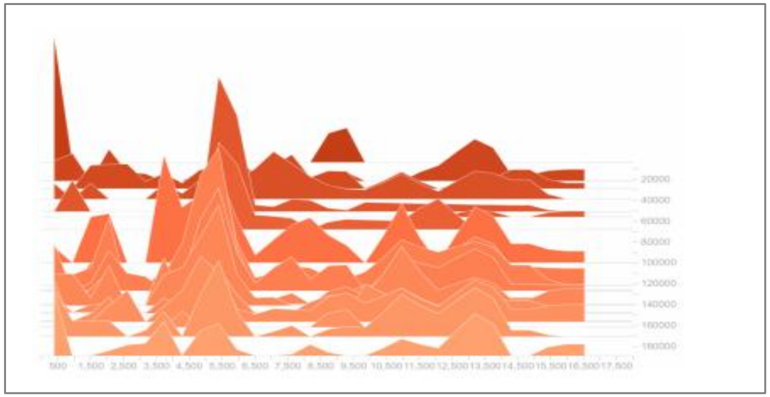
$$c_j \leftarrow \sum_{i=1}^{N} \mathbb{1}[z_i = j]; \quad \mu^j \leftarrow \frac{1}{c_j} \sum_{i=1}^{N} \mathbb{1}[z_i = j] x_i.$$

1 Introduction

Index Collapse



Index Collapse



Ideal Case

- X axis corresponds to the different possible discrete latent codes, Y axis corresponds to the progression of training steps.

- Only few latent embedding vectors are selected, and updated.

Figure from *Fast Decoding in Sequence Models Using Discrete Latent Variables, Lukasz et al.*

VQ-VAE training with EM

Instead of indexing

$$P_{\Theta}(z_i \mid z_e(x_i)) \propto e^{-\|e_{z_i} - z_e(x_i)\|_2^2}$$

Define probability distribution over embedding vectors

$$z_i^1, \cdots, z_i^m \sim \text{Multinomial}\left(-\|e_1 - z_e(x_i)\|_2^2, \ldots, -\|e_K - z_e(x_i)\|_2^2\right)$$

Monte Carlo Approximate

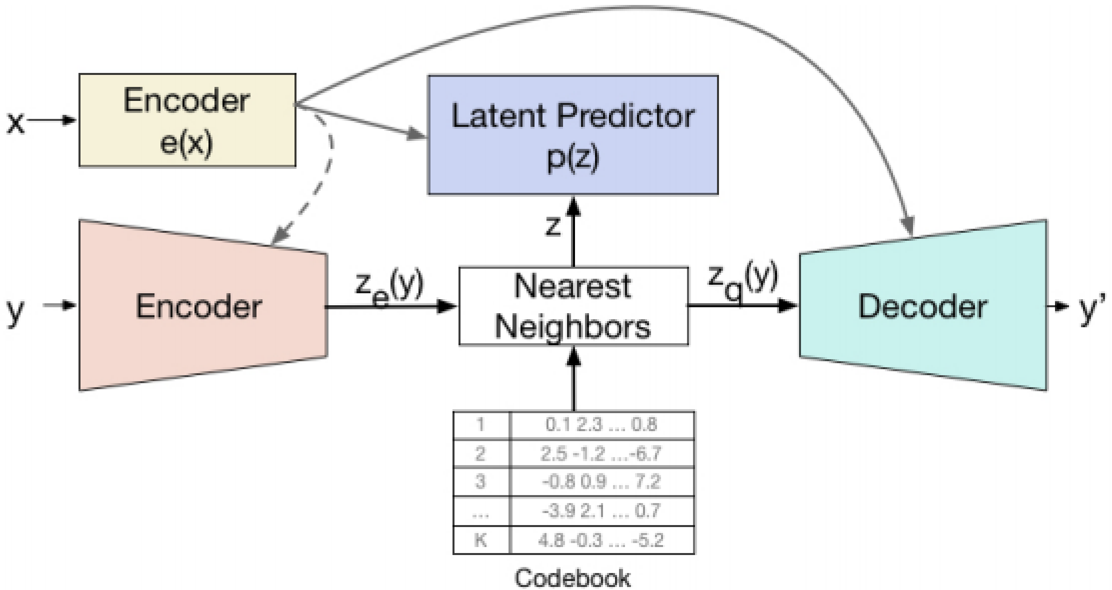
E step: $$z_i^1, \ldots, z_i^m \leftarrow \text{Multinomial}\left(-\|e_1 - z_e(x_i)\|_2^2, \ldots, -\|e_K - z_e(x_i)\|_2^2\right)$$

M step: $$c_j \leftarrow \frac{1}{m}\sum_{i=1}^{N}\sum_{l=1}^{m} \mathbb{1}\left[z_i^l = j\right]; \qquad e_j \leftarrow \frac{1}{mc_j}\sum_{i=1}^{N}\sum_{l=1}^{m} \mathbb{1}\left[z_i^l = j\right] z_e(x_i).$$

$$z_q(x_i) = \frac{1}{m}\sum_{l=1}^{m} e_{z_i^l}.$$

3 Experiments

Machine Translation



- Encoder function is a series of convolutional layers with residual connections
- Source sentence is encoded in to sequence of hidden states through multiple causal self-attention layers
- Decoder consists of transpose convolutional layers whose output is fed to a transformer decoder with causal attention.

3 Experiments

Machine Translation

Model	n_c	n_s	BLEU	Latency	Speedup
Autoregressive Model (beam size=4)	-	-	28.1	331 ms	1×
Autoregressive Baseline (no beam-search)	-	-	27.0	265 ms	1.25×
NAT + distillation	-	-	17.7	39 ms	15.6× *
NAT + distillation + NPD=10	-	-	18.7	79 ms	7.68× *
NAT + distillation + NPD=100	-	-	19.2	257 ms	2.36× *
LT + Semhash	-	-	19.8	105 ms	3.15×
Our Results					
VQ-VAE	3	-	21.4	81 ms	4.08×
VQ-VAE with EM	3	5	22.4	81 ms	4.08×
VQ-VAE + distillation	3	-	26.4	81 ms	4.08×
VQ-VAE with EM + distillation	3	10	**26.7**	81 ms	4.08×
VQ-VAE with EM + distillation	4	10	25.4	58 ms	5.71×

3 Experiments

Image Generation

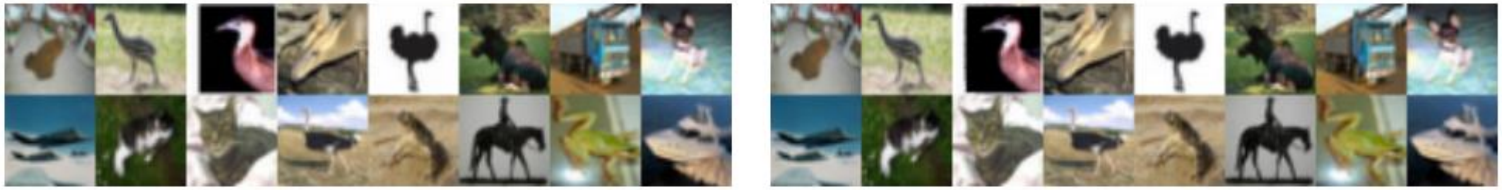


Figure 4: Samples of original and reconstructed images from CIFAR-10 using VQ-VAE trained using EM with a code-book of size 2^8.

Model	n_s	Log perplexity
ImageTransformer	-	2.92
VAE	-	4.51
VQ-VAE [31]	-	**4.67**
VQ-VAE (Ours)	-	4.83
EM	5	**4.80**