Unsupervised Learning

May 21st, 2019

Applied Deep Learning

YUN-NUNG (VIVIAN) CHEN

HTTP://ADL.MIULAB.TW

國立臺灣大學
National Taiwan University

Slide credit from Hung-Yi Lee

Introduction

Big data ≠ Big annotated data

Machine learning techniques include:

- Supervised learning (if we have labelled data)
- Reinforcement learning (if we have an environment for reward)
- Unsupervised learning (if we do not have labelled data)

What can we do if there is no sufficient training data?

Semi-Supervised Learning

Labelled data

cat

dog

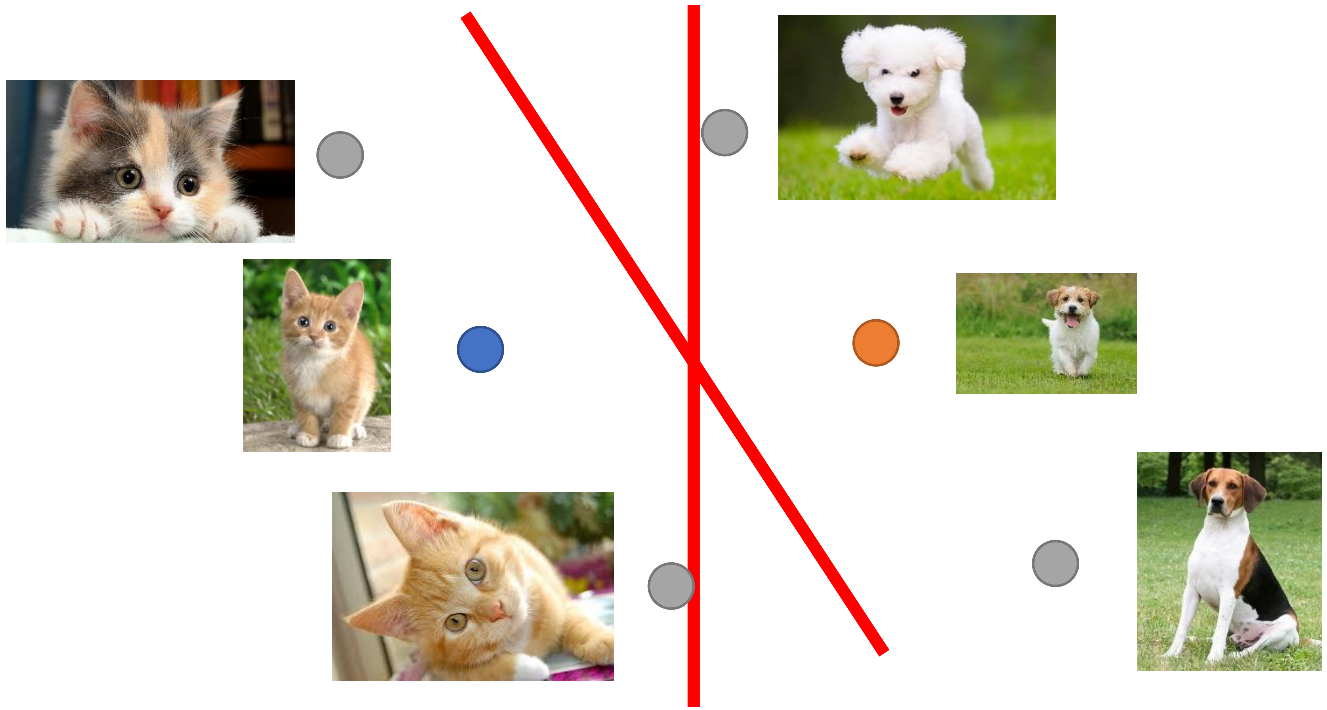
Unlabeled data




(Image of cats and dogs without labeling)

Semi-Supervised Learning

Why semi-supervised learning helps?



The distribution of the unlabeled data provides some cues

Transfer Learning

Labelled data

cat

dog

Labeled data

elephant

elephant

tiger

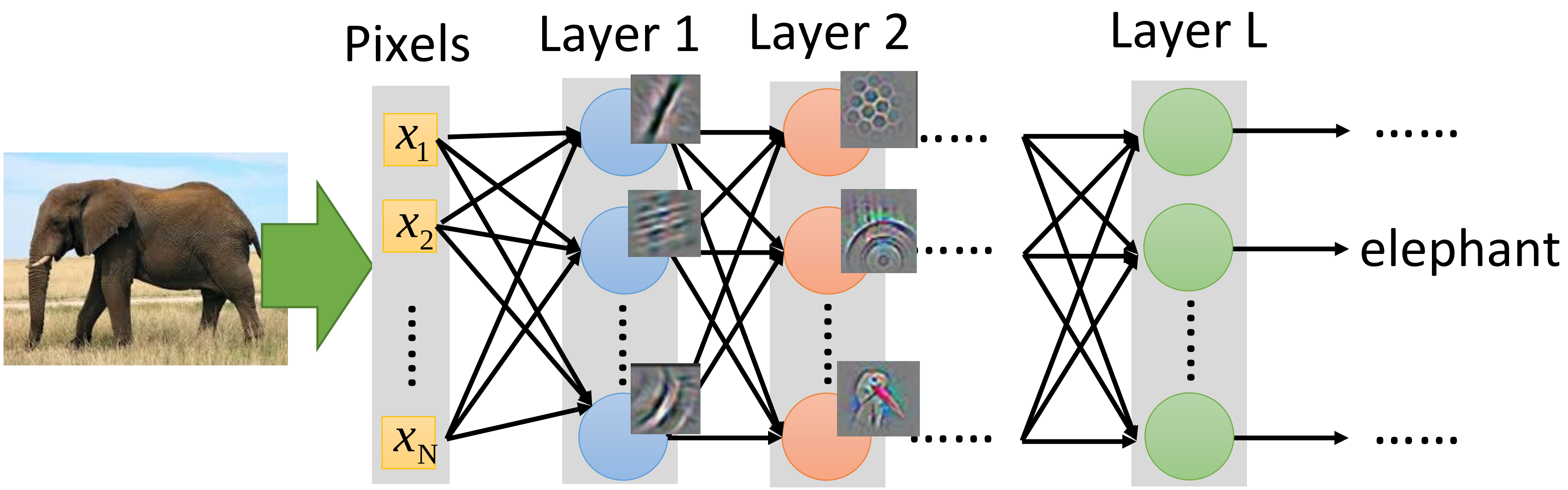
tiger

Not related to the task considered

Transfer Learning

Widely used on image processing

- Using sufficient labeled data to learn a CNN
- Using this CNN as feature extractor

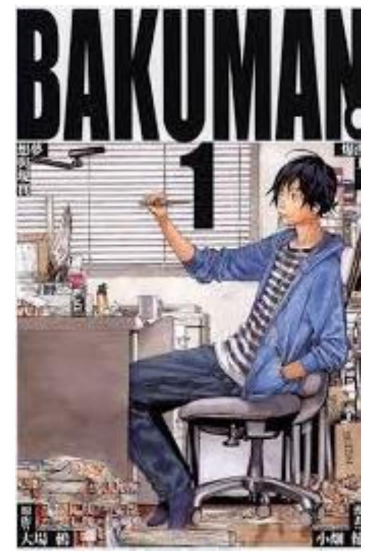


Transfer Learning Example

<u>研究生 online</u> ***<u>漫畫家 online</u>***

研究生生存守則

研究生生存守則		漫畫家 online
研究生	←	漫畫家
指導教授	←	責編
跑實驗	←	畫分鏡
投稿期刊	←	投稿 jump



爆漫王

Self-Taught Learning

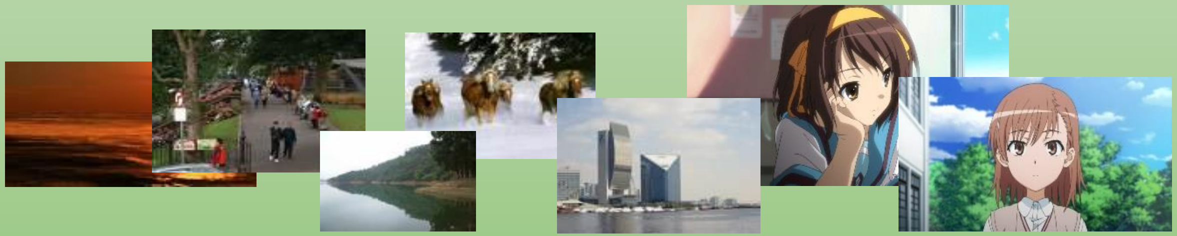
The unlabeled data sometimes is not related to the task

Labelled data

cat

dog

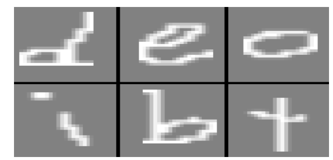
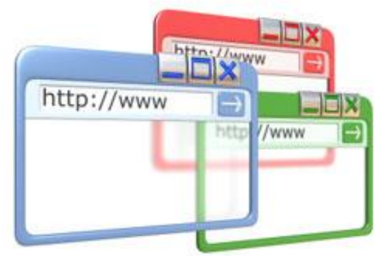
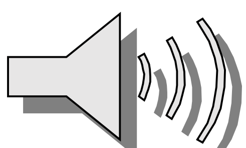
Unlabeled data



(Just crawl millions of images from the Internet)

Self-Taught Learning

The unlabeled data sometimes is not related to the task

	Labelled data	Unlabeled data
Digit Recognition	 Digits	 character
Document Classification	 News	 Webpages
Speech Recognition	 Taiwanese	YouTube English Chinese ……

Why can we use unlabeled and unrelated data to help our tasks?

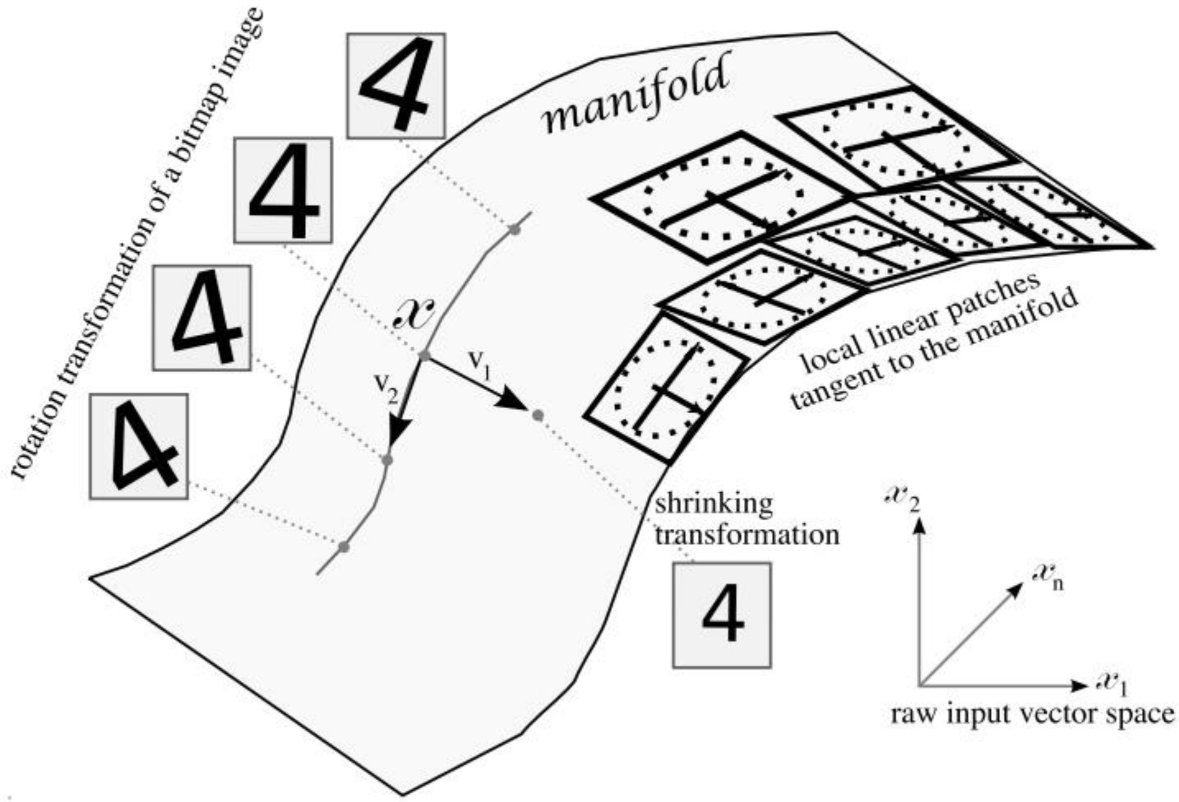
Self-Taught Learning

How does self-taught learning work?

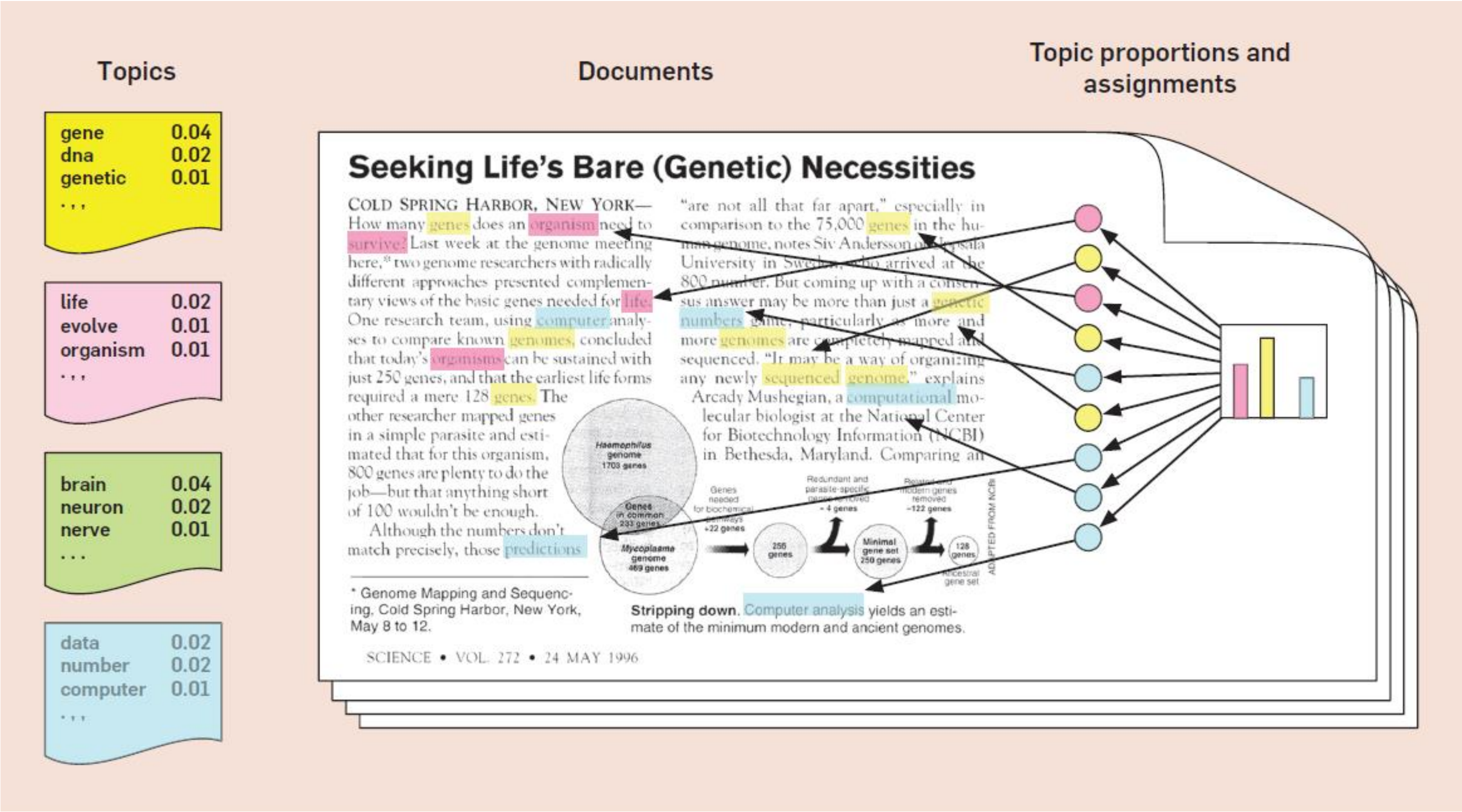
Why does unlabeled and unrelated data help the tasks?

Finding latent factors that control the observations

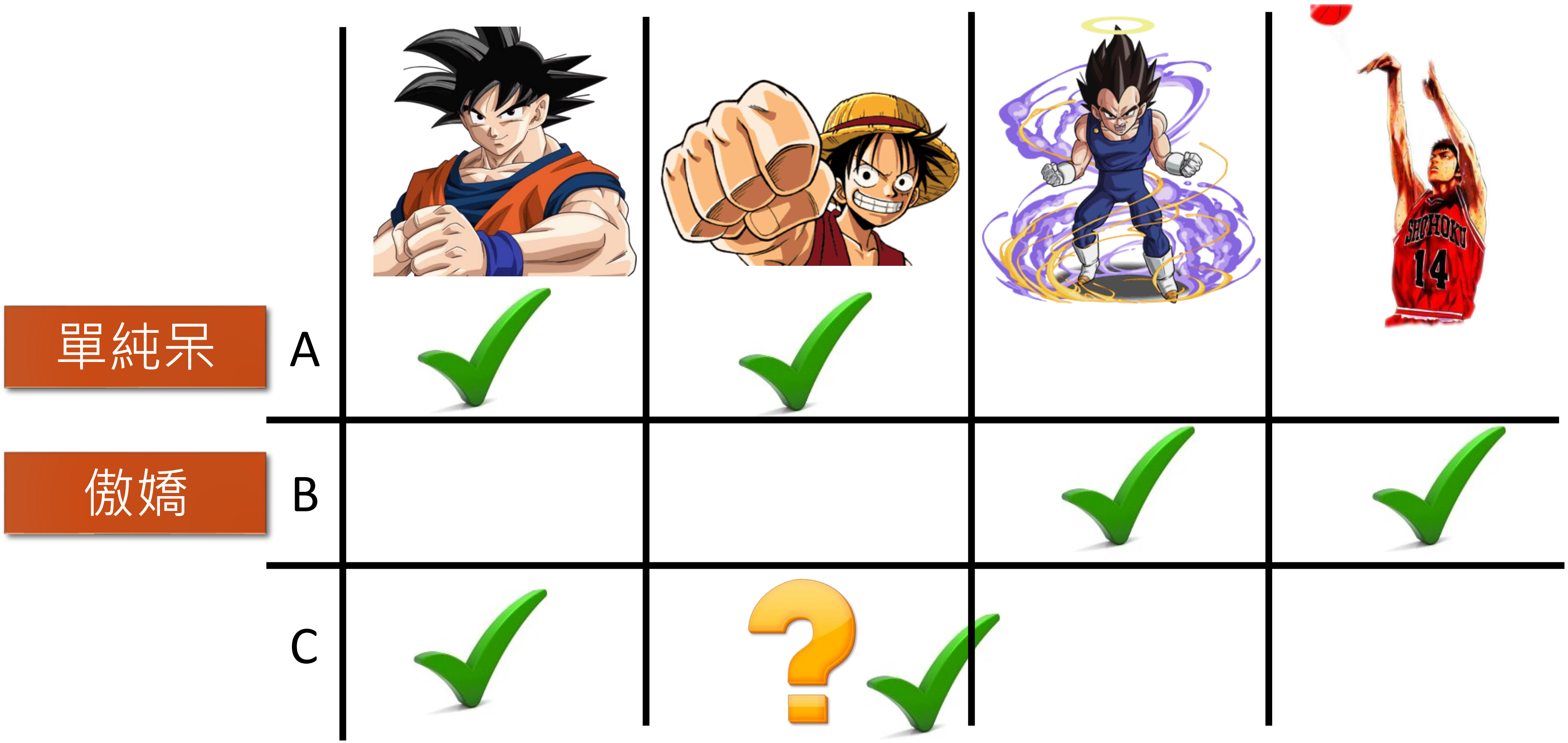
Latent Factors for Handwritten Digits



Latent Factors for Documents



Latent Factors for Recommendation System

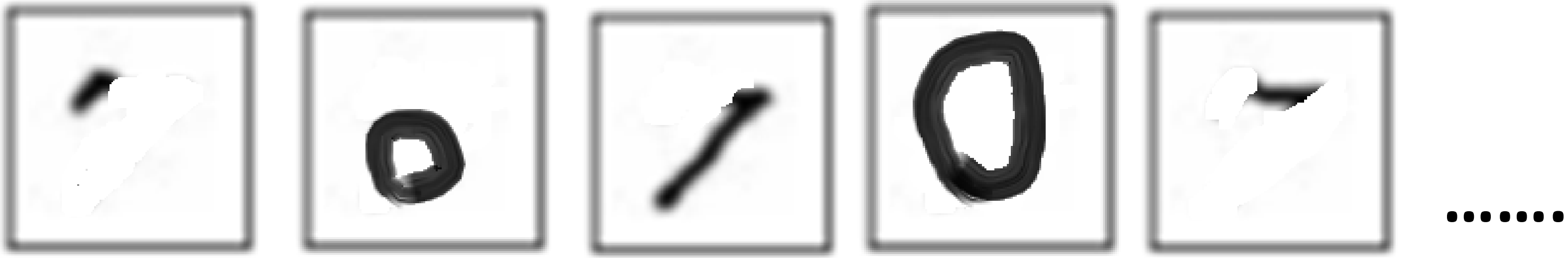


Latent Factor Exploitation

Handwritten digits

The handwritten images are composed of **strokes**

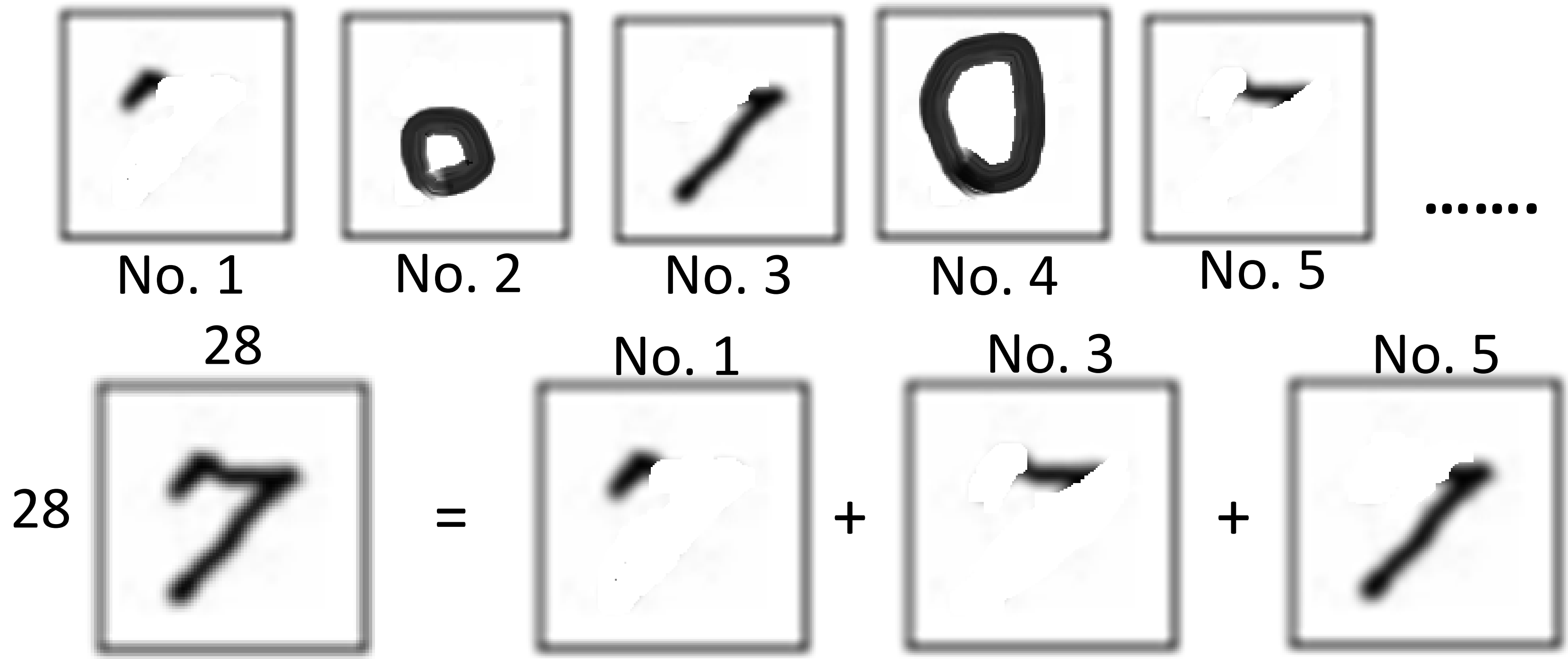
<u>Strokes (Latent Factors)</u>



No. 1 No. 2 No. 3 No. 4 No. 5

Latent Factor Exploitation

<u>Strokes (Latent Factors)</u>



Represented by
28 X 28 = 784 pixels

[1 0 1 0 1 0 ....]
(simpler representation)

Autoencoder

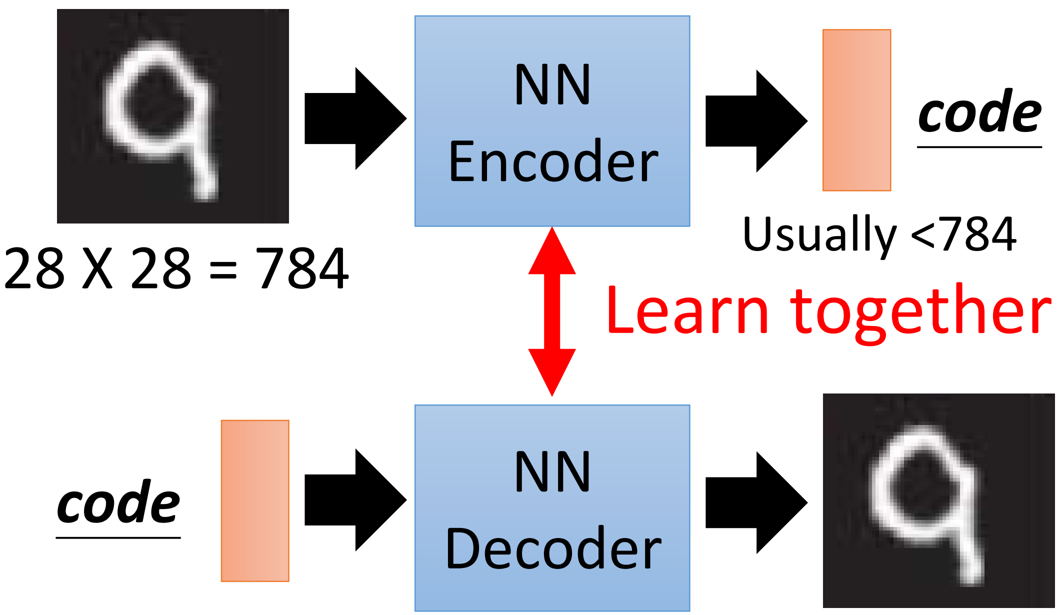
Representation Learning

Autoencoder

Represent a digit using 28 X 28 dimensions

Not all 28 X 28 images are digits

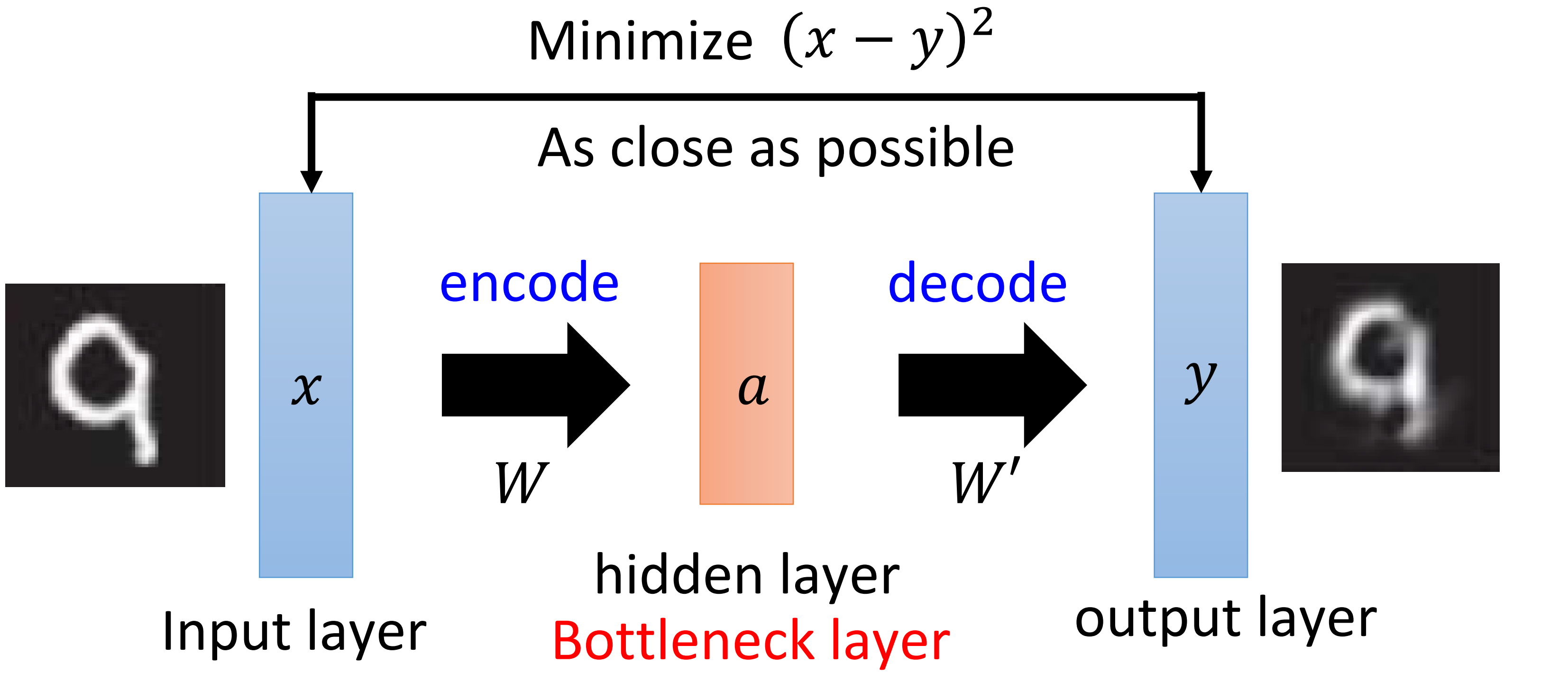
Idea: represent the images of digits in a more compact way



Compact representation of the input object

Can reconstruct the original object

Autoencoder

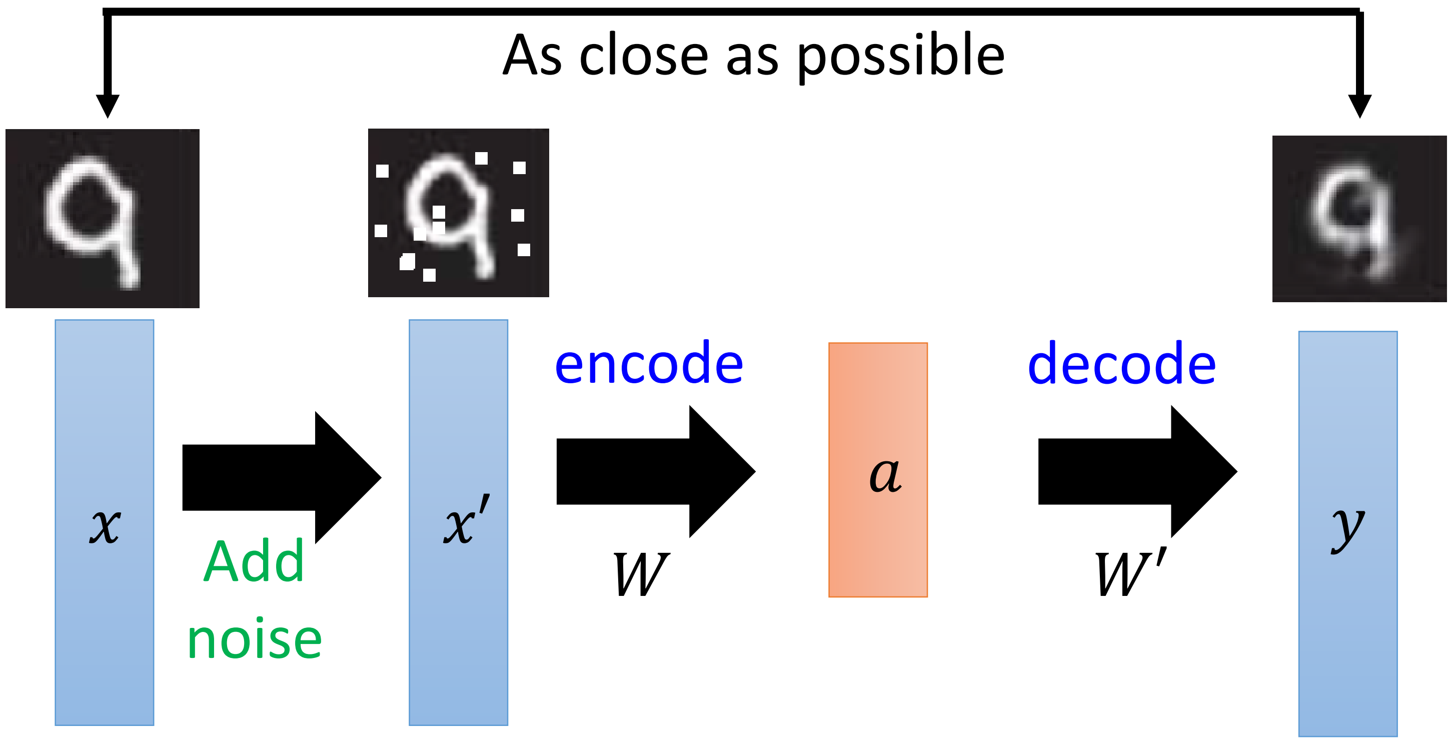


$$a = \sigma(Wx + b) \qquad y = \sigma(W'a + b')$$

Output of the hidden layer is the code

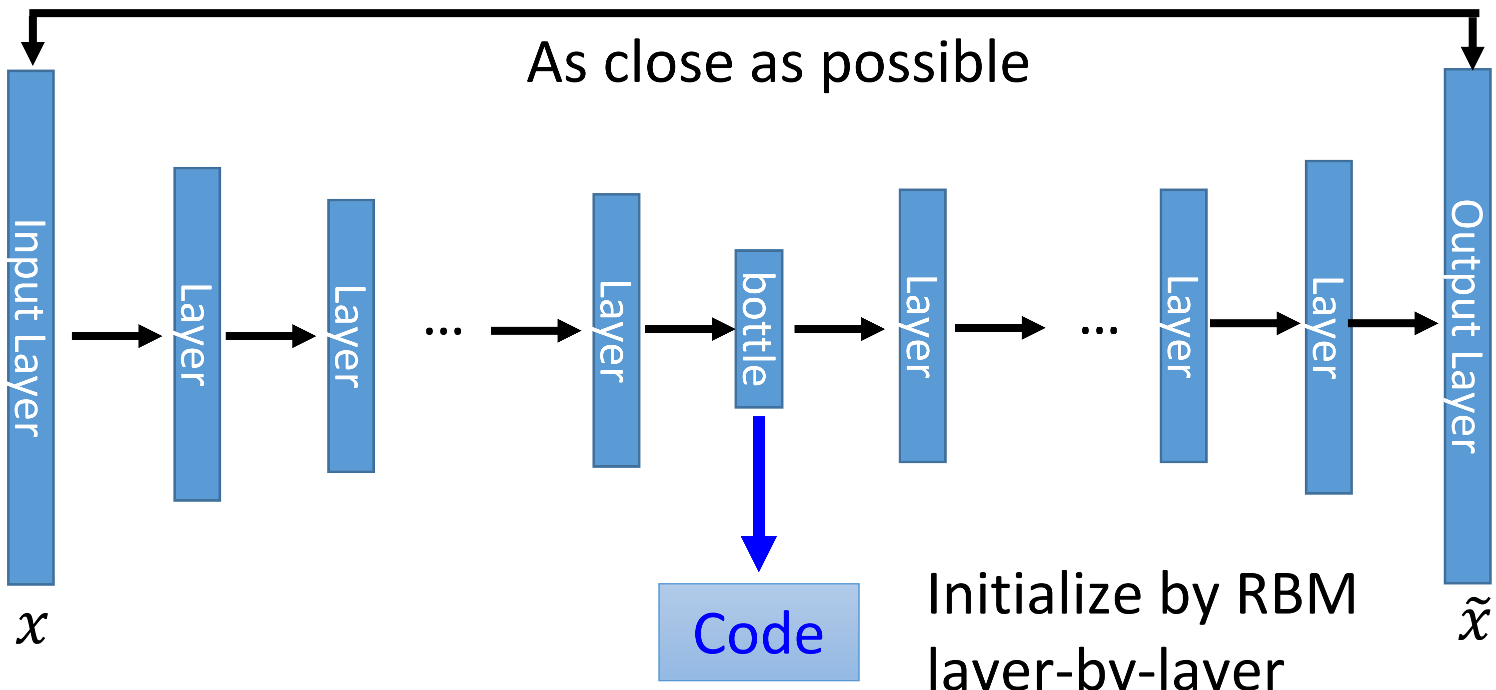
Autoencoder

De-noising auto-encoder



Rifai, et al. "Contractive auto-encoders: Explicit invariance during feature extraction,“ in *ICML*, 2011.

Deep Autoencoder



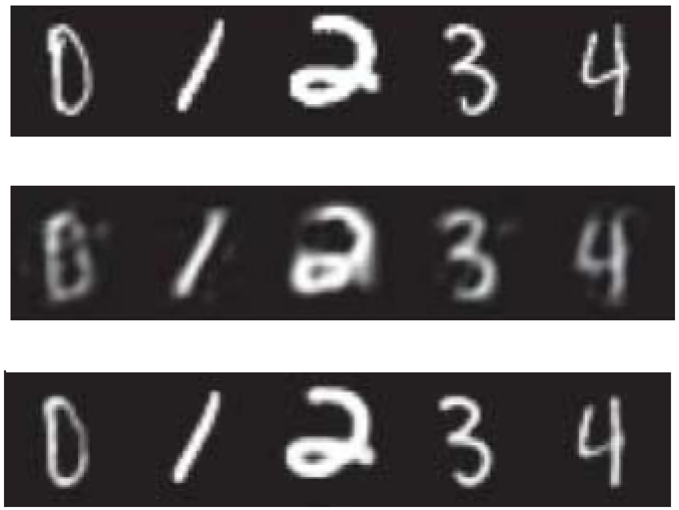
Initialize by RBM layer-by-layer

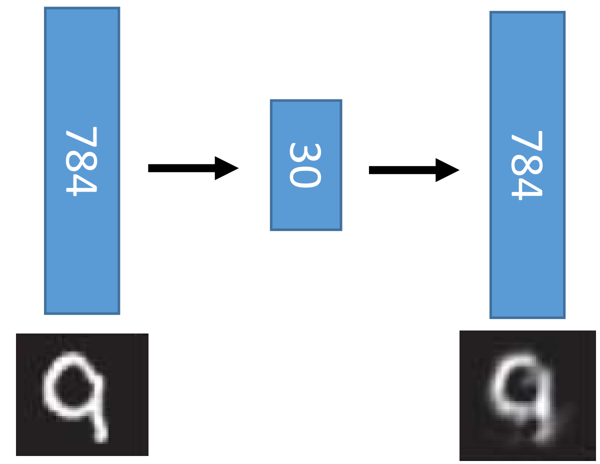
Deep Autoencoder

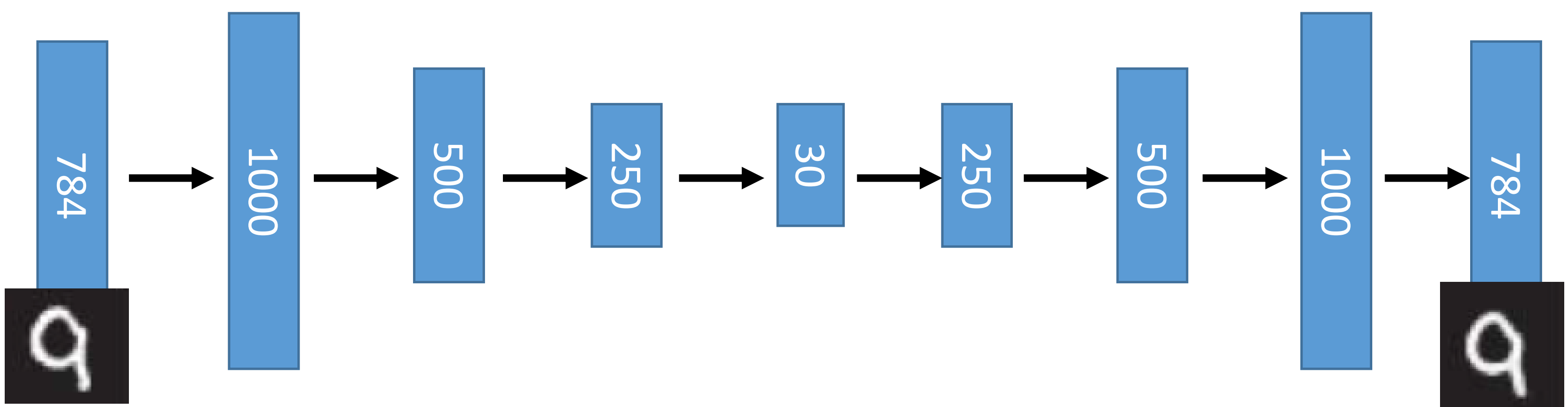
Original Image

PCA

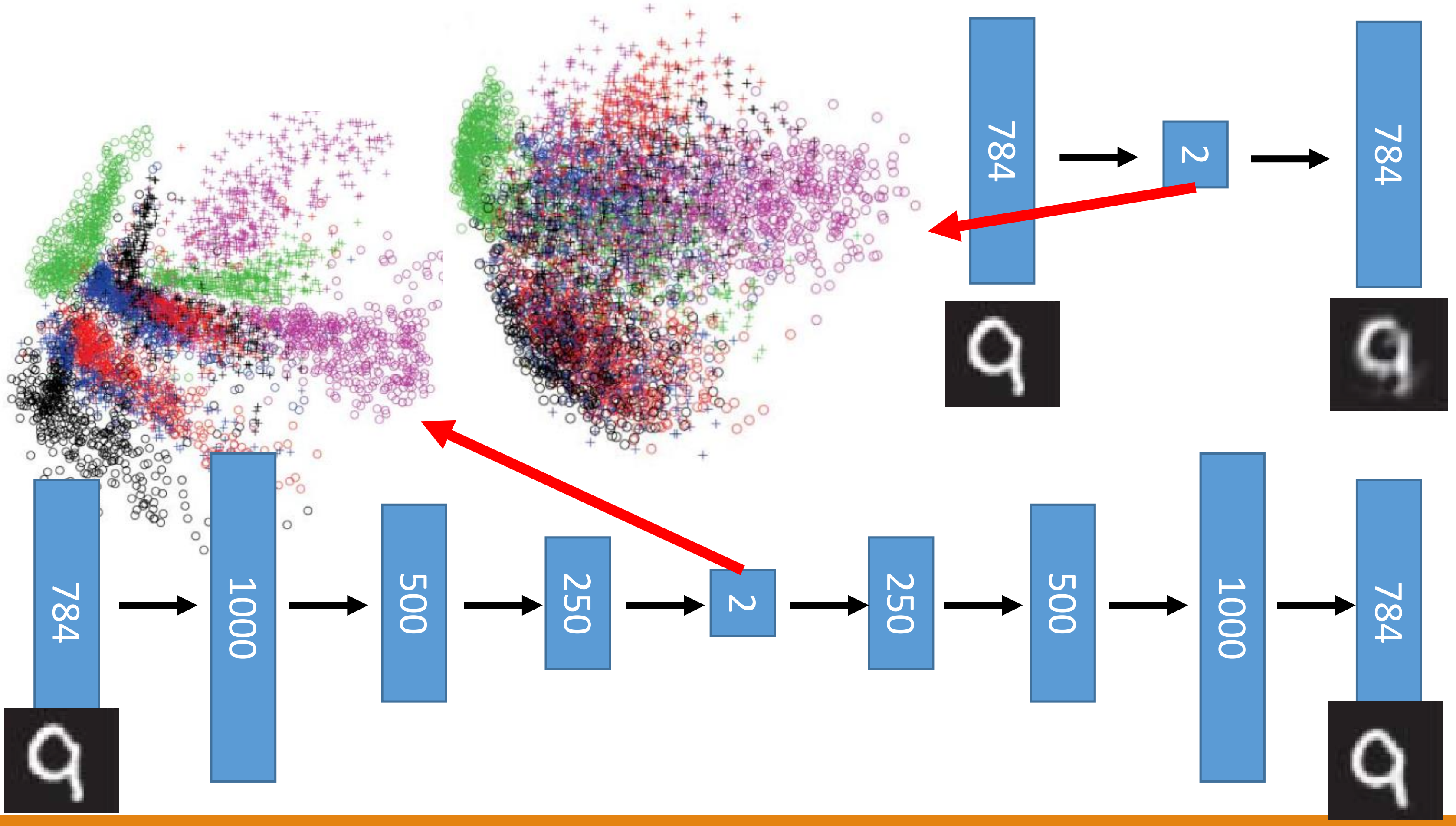
Deep Auto-encoder





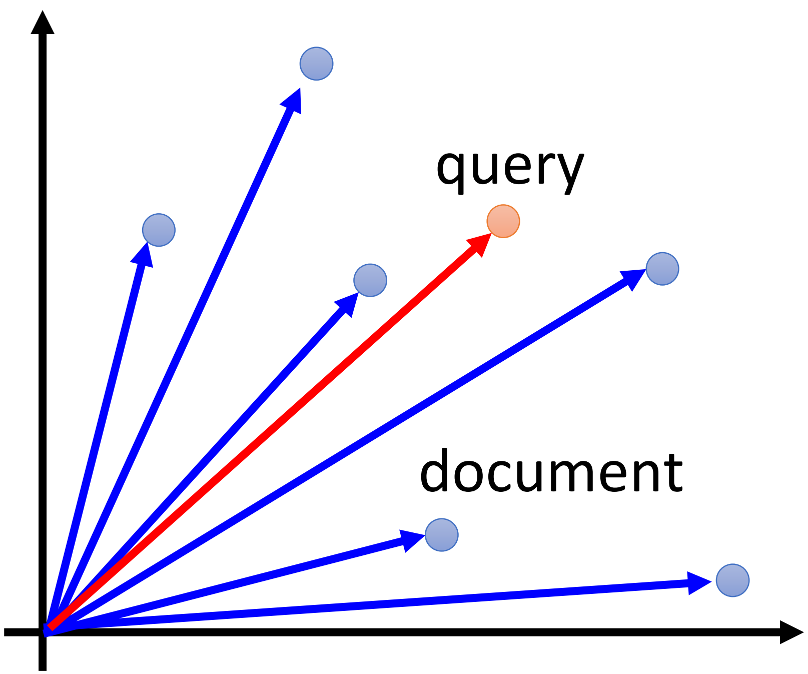


Feature Representation



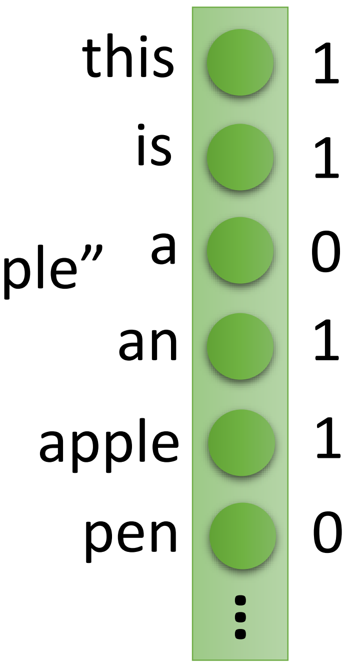
Auto-encoder – Text Retrieval

Vector Space Model



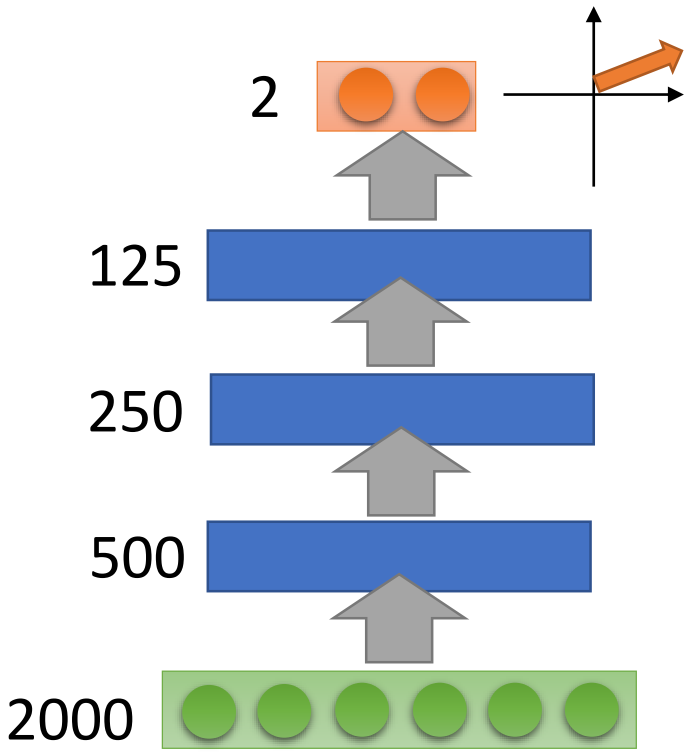
Bag-of-word

word string:
“This is an apple”

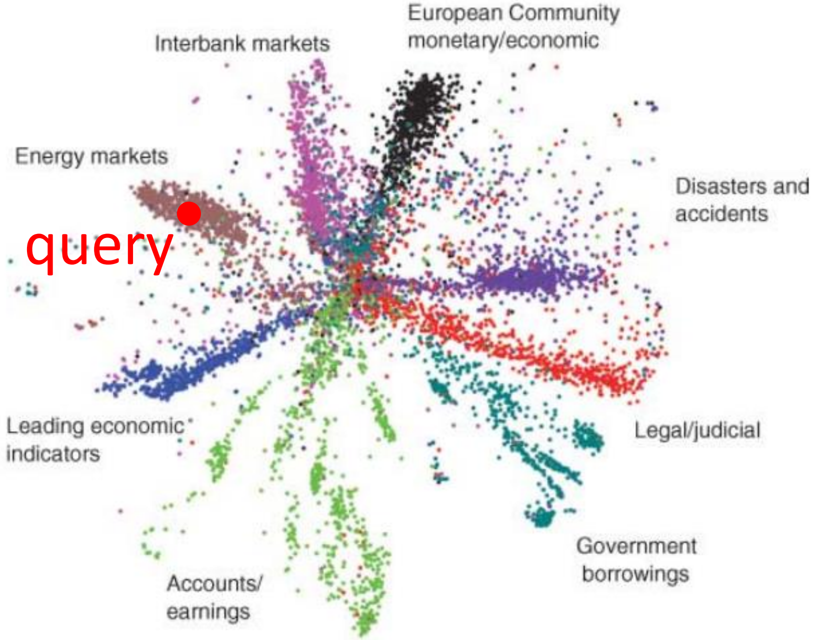


Semantics are not considered

Autoencoder – Text Retrieval



Bag-of-word (document or query)

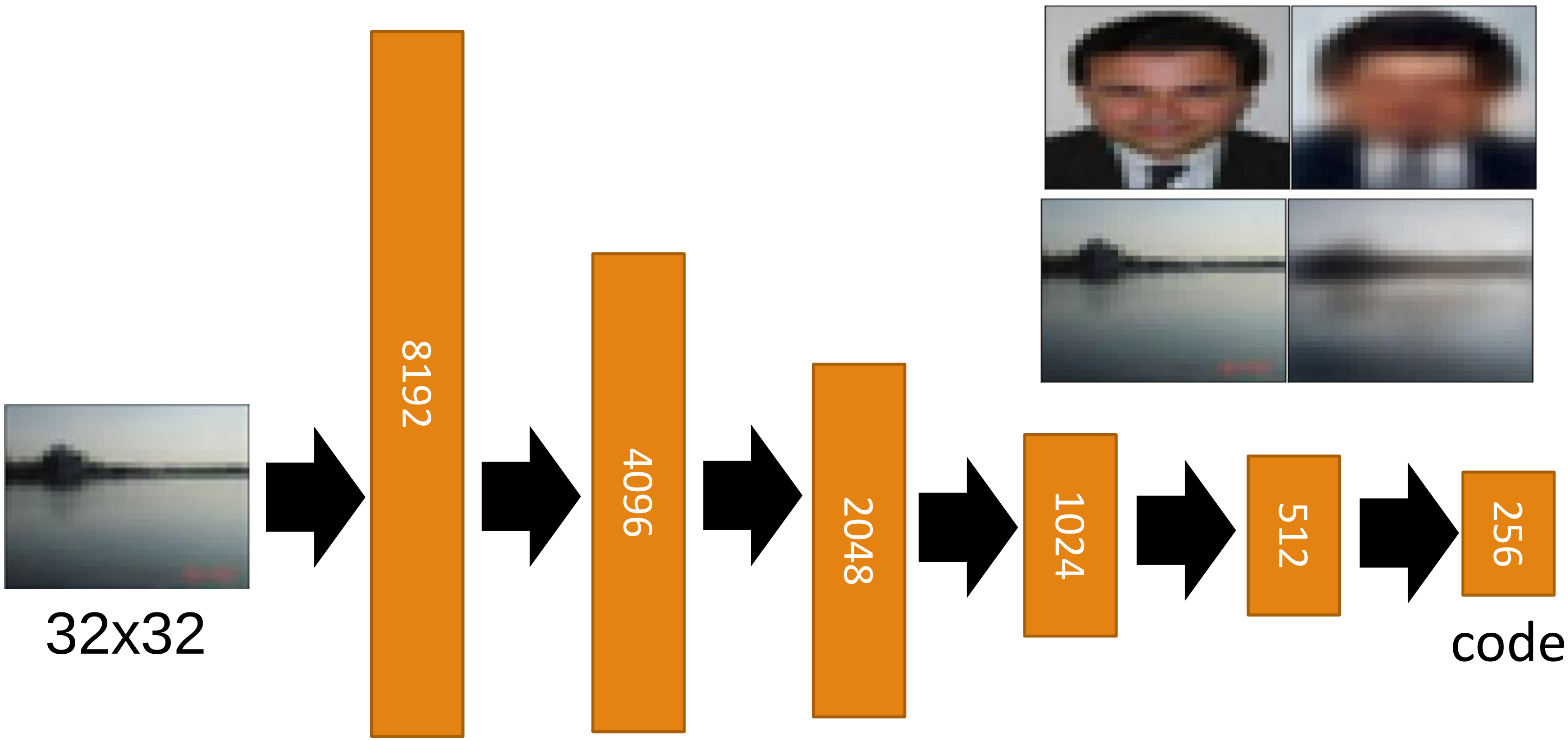


The documents talking about the same thing will have close code

Autoencoder – Similar Image Retrieval

Retrieved using Euclidean distance in pixel intensity space

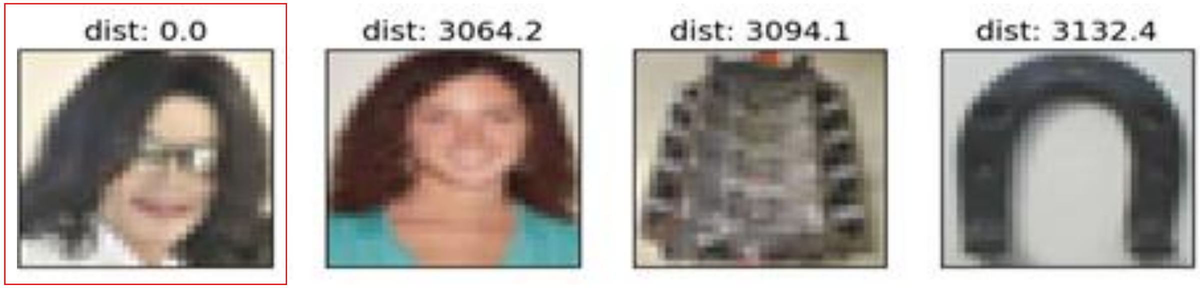
Autoencoder – Similar Image Retrieval



(crawl millions of images from the Internet)

Autoencoder – Similar Image Retrieval

Images retrieved using Euclidean distance in pixel intensity space

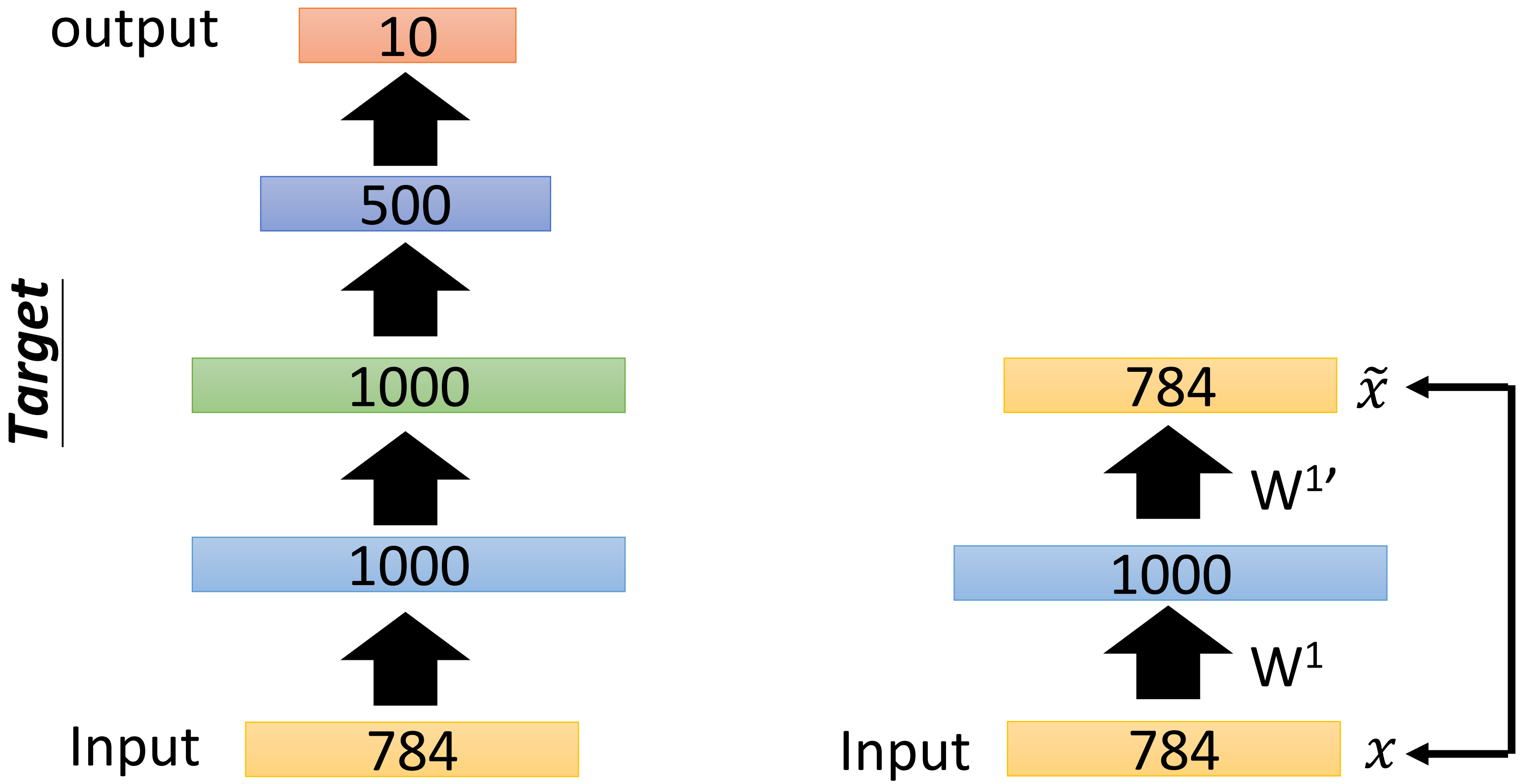


Images retrieved using 256 codes

Learning the useful latent factors

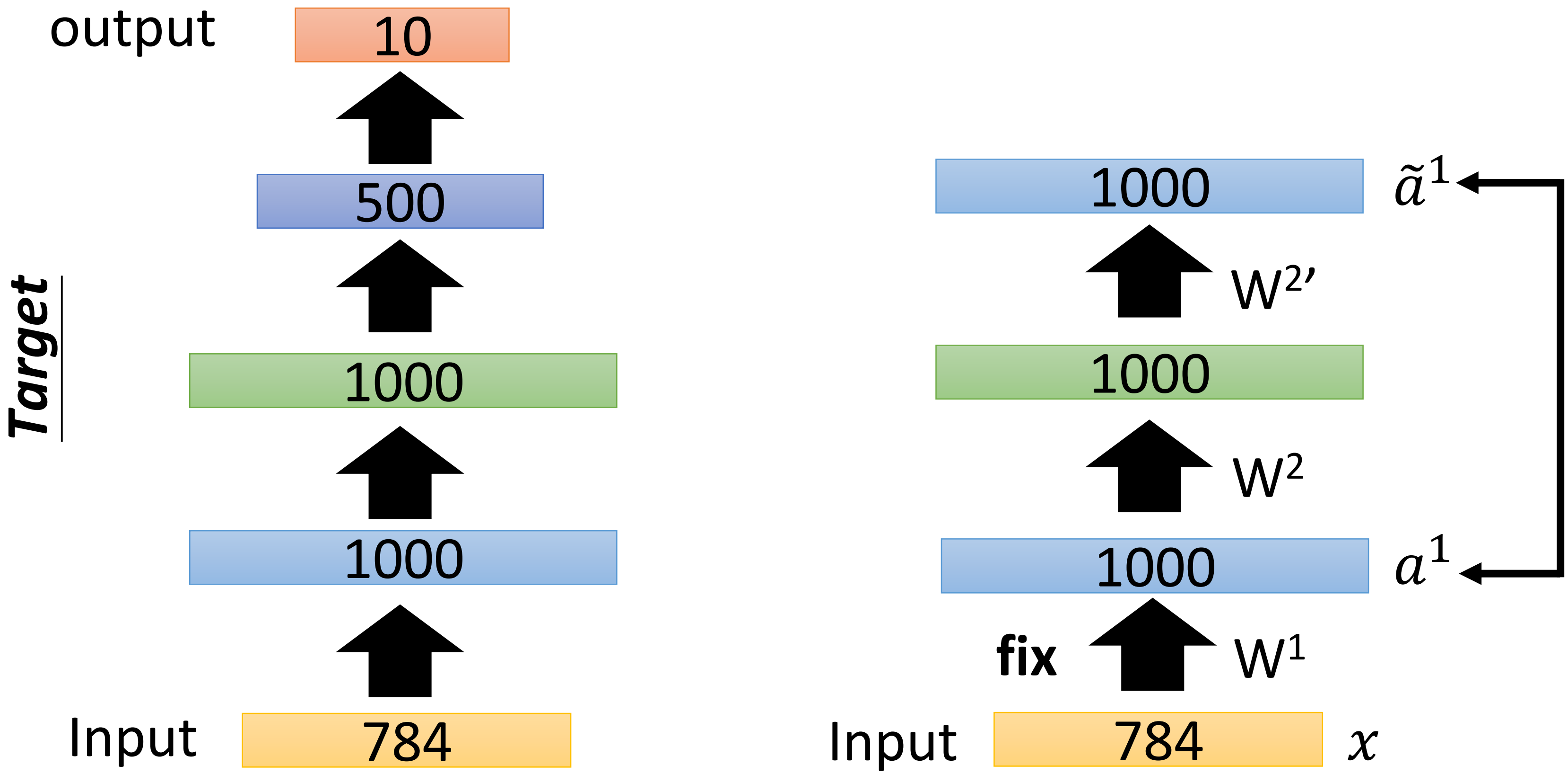
Autoencoder for DNN Pre-Training

Greedy layer-wise pre-training *again*



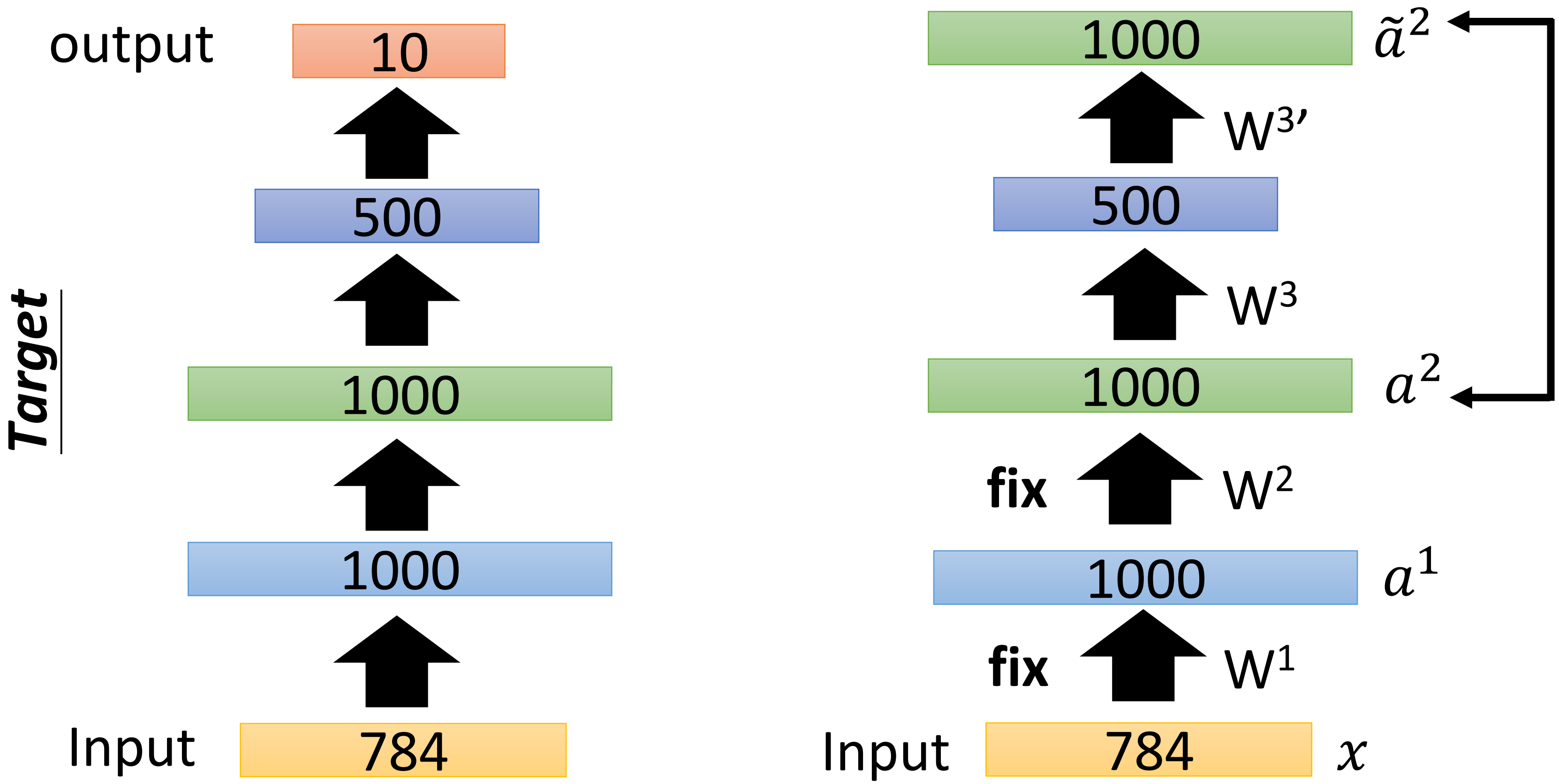
Autoencoder for DNN Pre-Training

Greedy layer-wise pre-training *again*



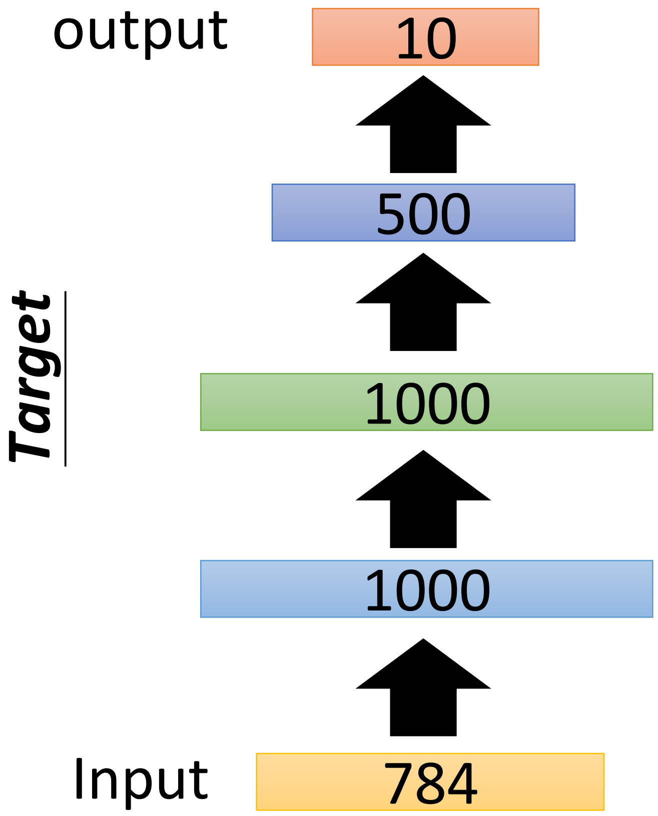
Autoencoder for DNN Pre-Training

Greedy layer-wise pre-training *again*

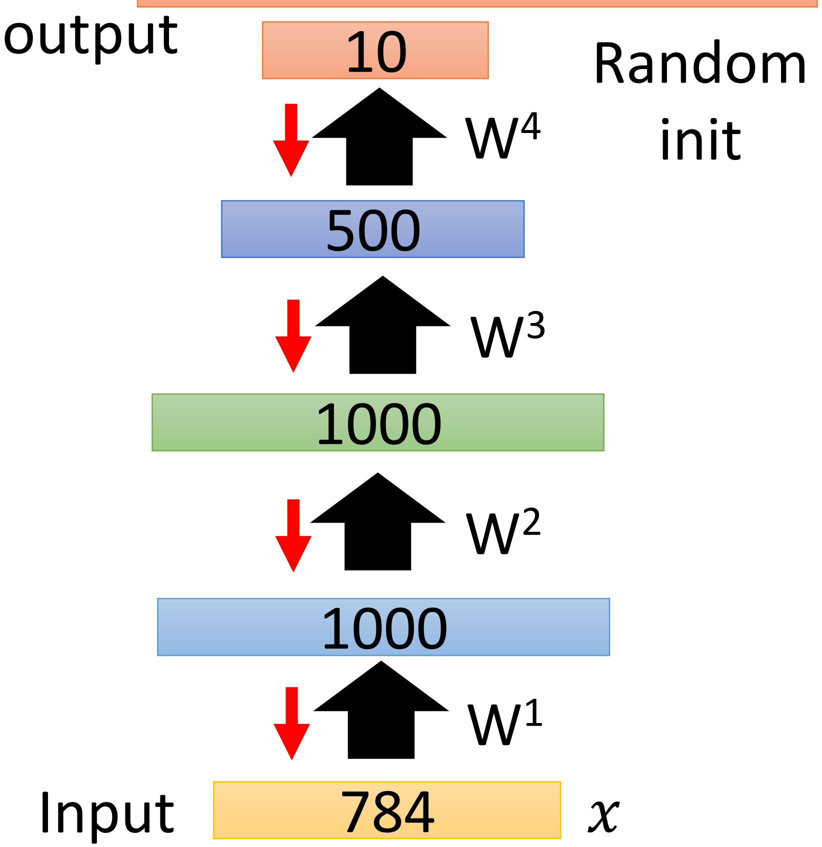


Autoencoder for DNN Pre-Training

Greedy layer-wise pre-training *again*



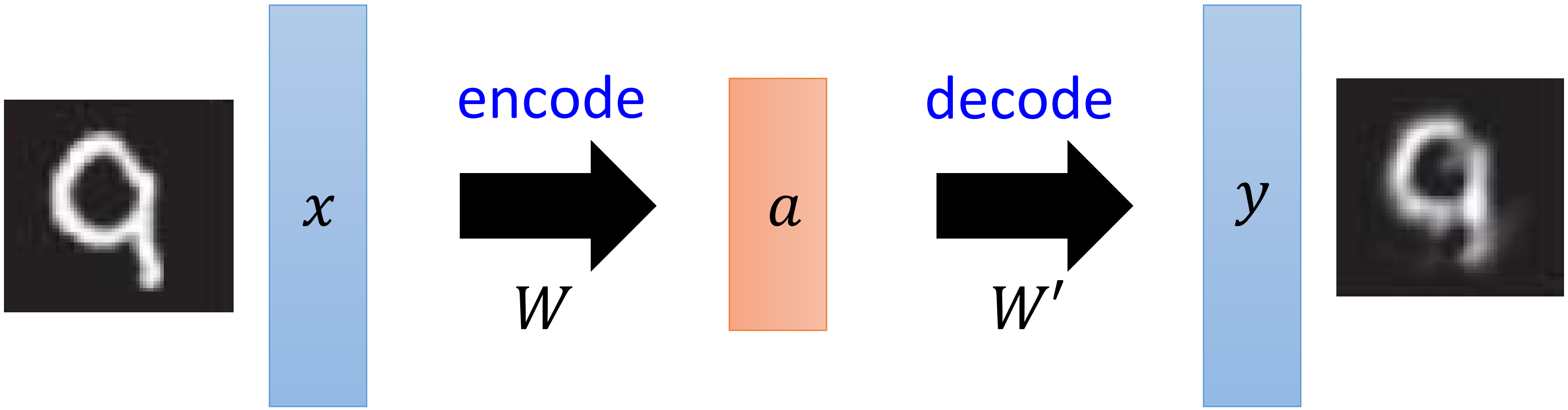
Find-tune via backprop



Variational Autoencoder

Representation Learning and Generation

Generation from Latent Codes

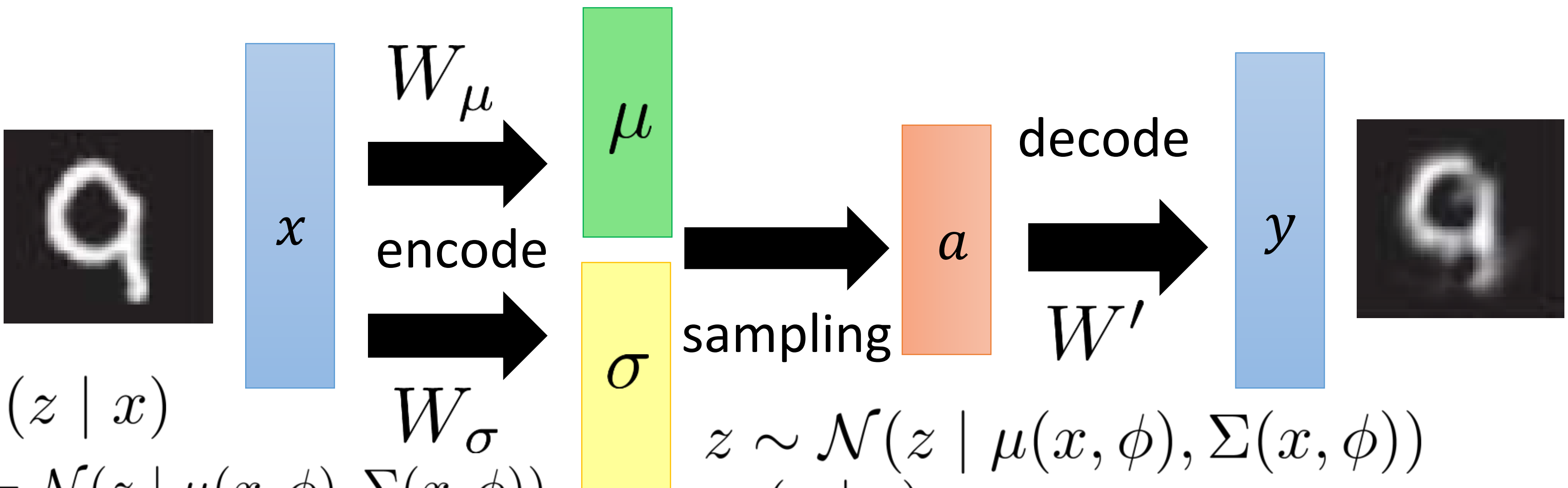


How can we set a latent code for generation?

Latent Code Distribution Constraints

Constrain the data distribution for learned latent codes

Generate the latent code via a prior distribution



$$q_\phi(z \mid x) = \mathcal{N}(z \mid \mu(x,\phi), \Sigma(x,\phi))$$

$$z \sim \mathcal{N}(z \mid \mu(x,\phi), \Sigma(x,\phi))$$

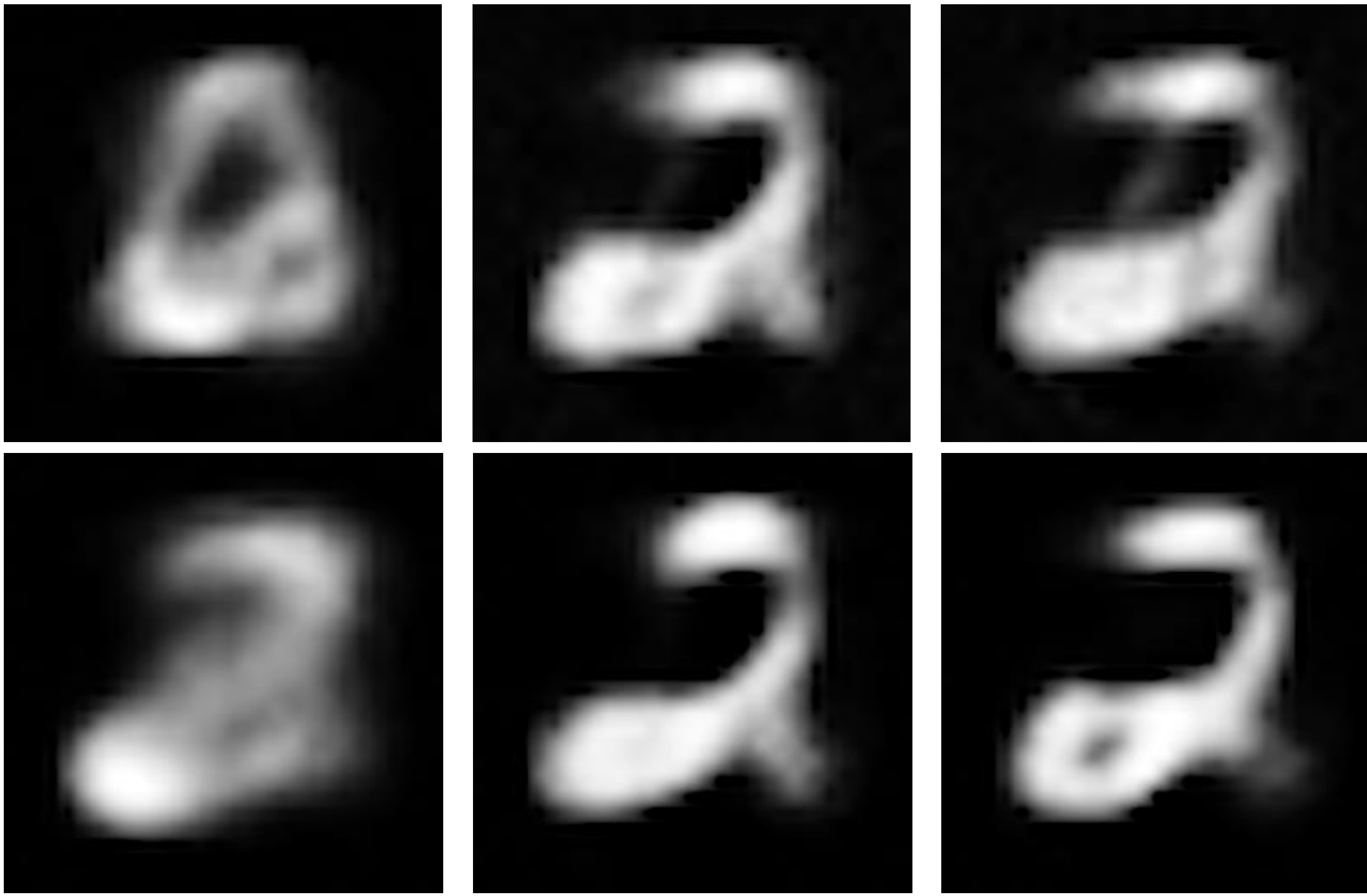
$$p_\theta(x \mid z)$$

$$\mathcal{L}(x) = \mathrm{E}_{q_\phi(z|x)}[\log p_\theta(x \mid z)] - D_{KL}(q_\phi(z \mid x) \| p_\theta(z))$$

Reconstruction

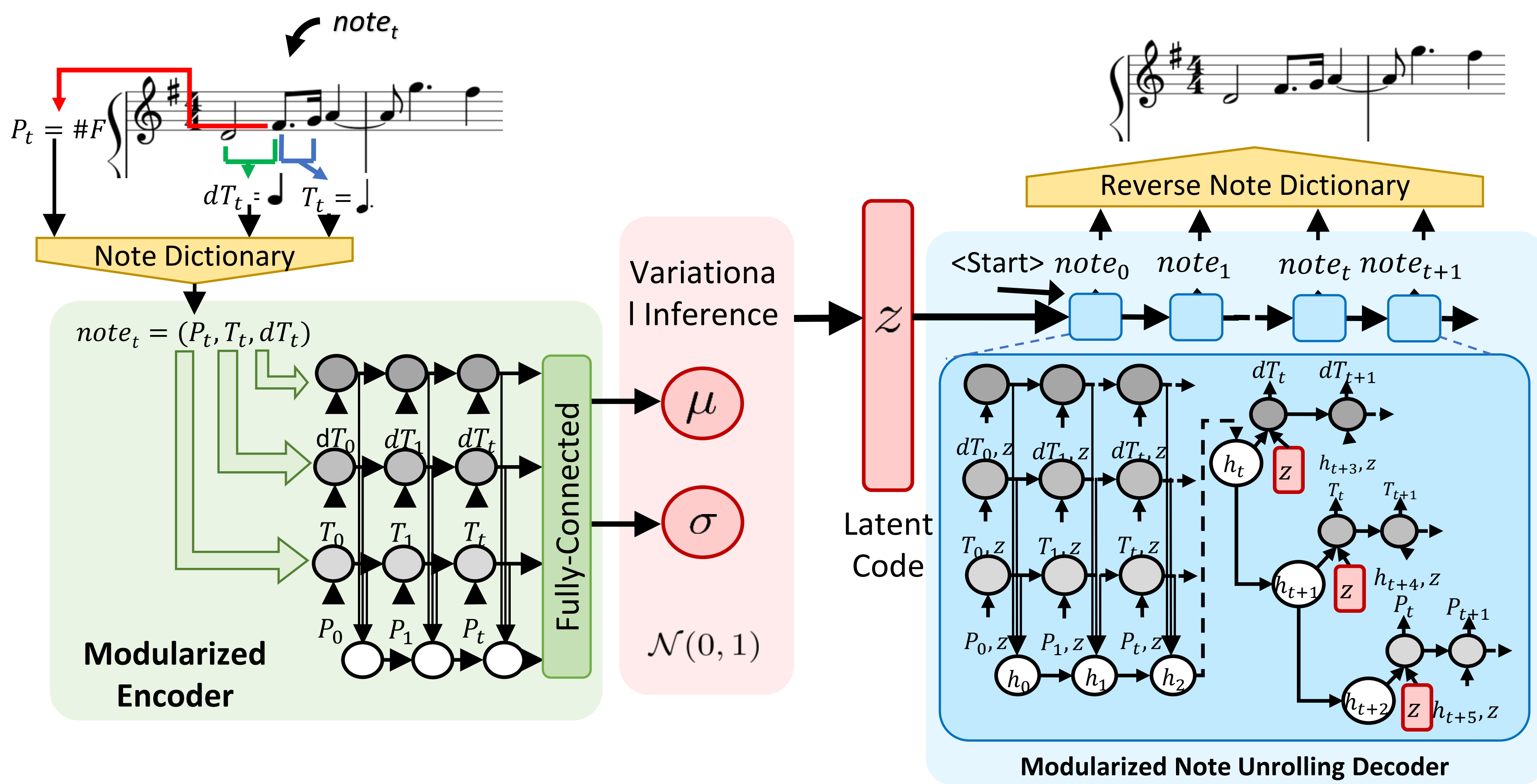
AE

VAE



VAE for Music Generation

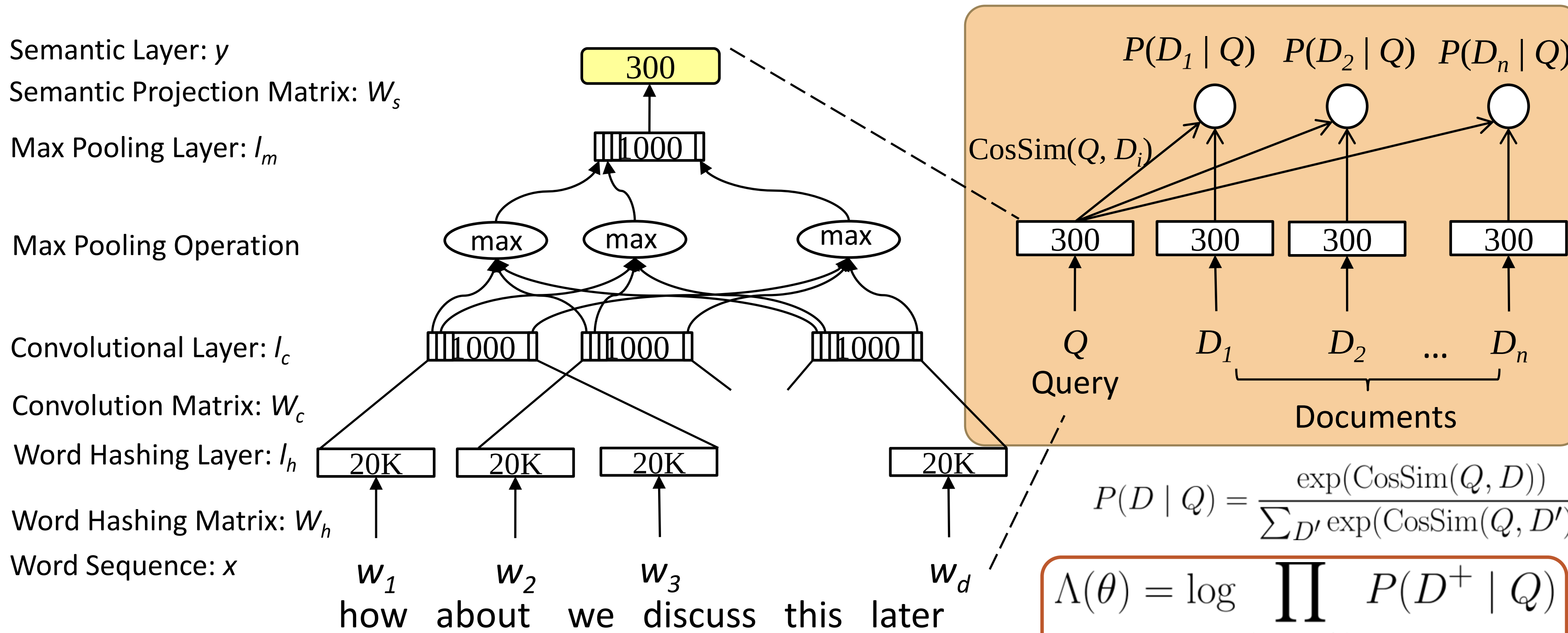
http://mvae.miulab.tw



Distant Supervision

Representation Learning by Weak Labels

Convolutional Deep Structured Semantic Models (CDSSM/DSSM)



$$P(D \mid Q) = \frac{\exp(\mathrm{CosSim}(Q, D))}{\sum_{D'} \exp(\mathrm{CosSim}(Q, D'))}$$

$$\Lambda(\theta) = \log \prod_{(Q,D^+)} P(D^+ \mid Q)$$

maximizes the likelihood of clicked documents given queries

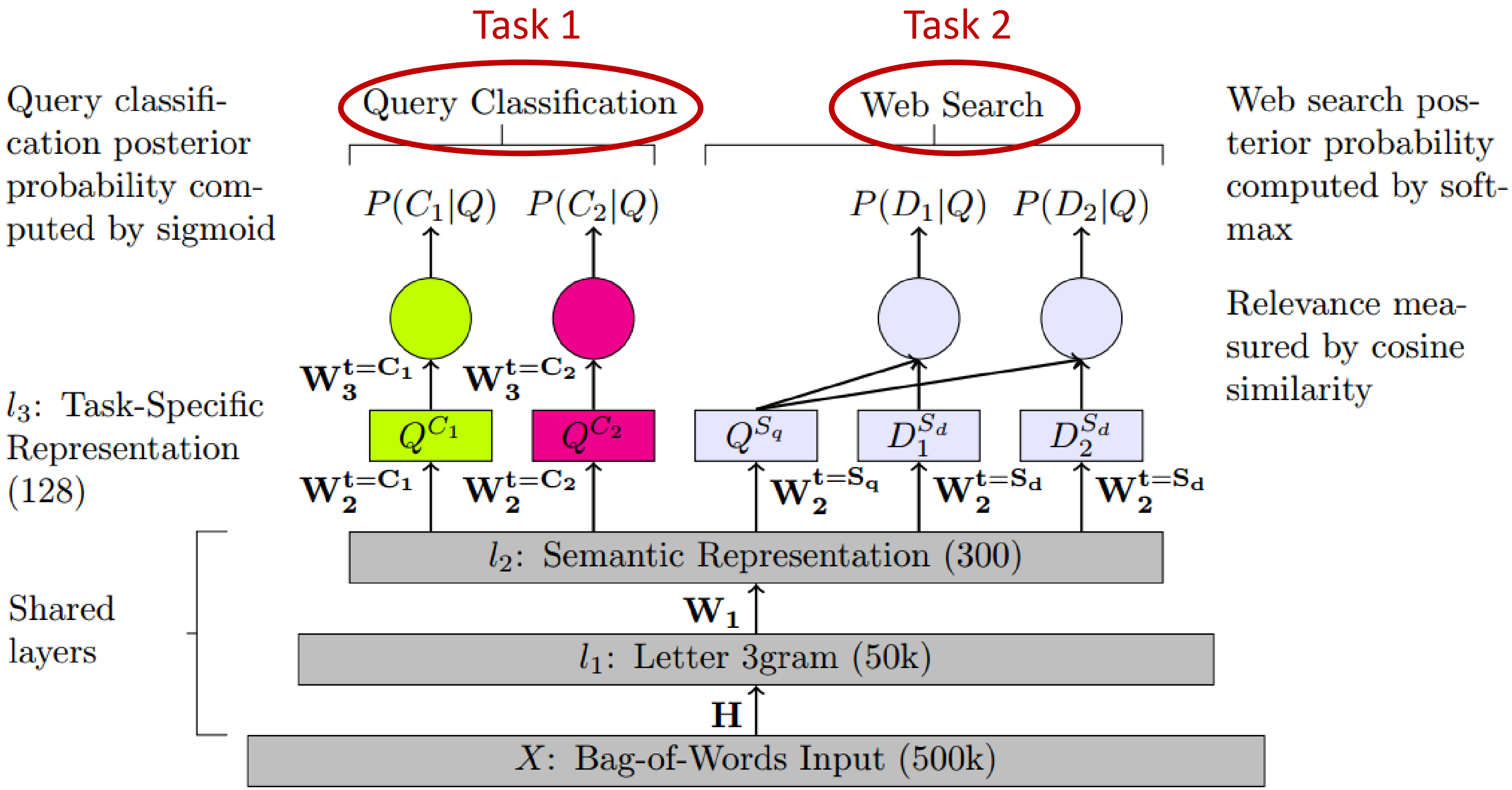
Semantically related documents are close to the query in the encoded space

Huang et al., "Learning deep structured semantic models for web search using clickthrough data," in *Proc. of CIKM*, 2013.
Shen et al., “Learning semantic representations using ´ convolutional neural networks for web search," in *Proc. of WWW*, 2014.

Multi-Tasking

Representation Learning by Different Tasks

Task-Shared Representation



The latent factors can be learned by different tasks

Concluding Remarks

Labeling data is expensive, but we have large unlabeled data

Autoencoder / VAE

- exploits unlabeled data to learn latent factors as representations
- learned representations can be transfer to other tasks

Distant Labels / Labels from Other Tasks

- learn the representations that are useful for other tasks
- learned representations may be also useful for the target task