Supervised Learning

Rowel Atienza, PhD

University of the Philippines

github.com/roatienza

2023

A computer program is said to learn from ***experience E*** *with respect to some class of* ***tasks T****, and* ***performance measure P****, if its performance at tasks in T, as measured by P, improves with experience E.* – Tom Mitchell

Supervised Learning

Experience

Task

Performance

Robot Learning

Learn from Experience (E) to perform Task (T) by improving the Performance (P)

https://spectrum.ieee.org/covariant-ai-gigantic-neural-network-to-automate-warehouse-picking

Supervised Learning

Experience: Dataset $\mathcal{D}_{train} = \{\boldsymbol{x}_n, y_n\}$ with N examples: input $\boldsymbol{x}_n \in \mathbb{R}^D$ with corresponding label $y_n \in \mathbb{R}$

Task: Classifier $y = f(\boldsymbol{x}, \boldsymbol{\theta}) : \mathbb{R}^D \rightarrow \mathbb{R}$ where $\boldsymbol{\theta}$ are parameters

Ground Truth Experience: Dataset $\mathcal{D}_{test} = \{\boldsymbol{x}_m, y_m\}$, $\hat{y}_m = f(\boldsymbol{x}_m, \boldsymbol{\theta})$ for $m = 1 \ldots M$

Performance: Accuracy on $\mathcal{D}_{test}$

$$Accuracy = \frac{\#\ Correct\ Prediction}{Size\ of\ Test\ Dataset} = \frac{\sum_{m=1}^{M}(\hat{y}_m\ equals\ y_m)}{M}$$

Machine Learning & Generalization

Initial Test $Accuracy_0$ of $\hat{y}_m = f(\boldsymbol{x}_m, \boldsymbol{\theta}^0)$, $\boldsymbol{\theta}^0$ initial parameters

Machine Learning: Finding the optimal parameters $\boldsymbol{\theta}^*$ using $\mathcal{D}_{train} = \{\boldsymbol{x}_n, y_n\}$: $\boldsymbol{\theta}^0 \rightarrow \boldsymbol{\theta}^*$

$$f(\boldsymbol{x}_n, \boldsymbol{\theta}^*) \approx y_n \quad n = 1 \ldots N$$

Final Test $Accuracy_*$ of $\hat{y}_m = f(\boldsymbol{x}_m, \boldsymbol{\theta}^*)$

Machine Learning: $Accuracy_* > Accuracy_0$

Generalization: $Accuracy_{*(train)} - Accuracy_{*(test)} = Gap \rightarrow 0$

Experience

The Datasets

MNIST Dataset

$\mathcal{D}_{train} = \{\boldsymbol{x}_n, y_n\}, N = 60{,}000$

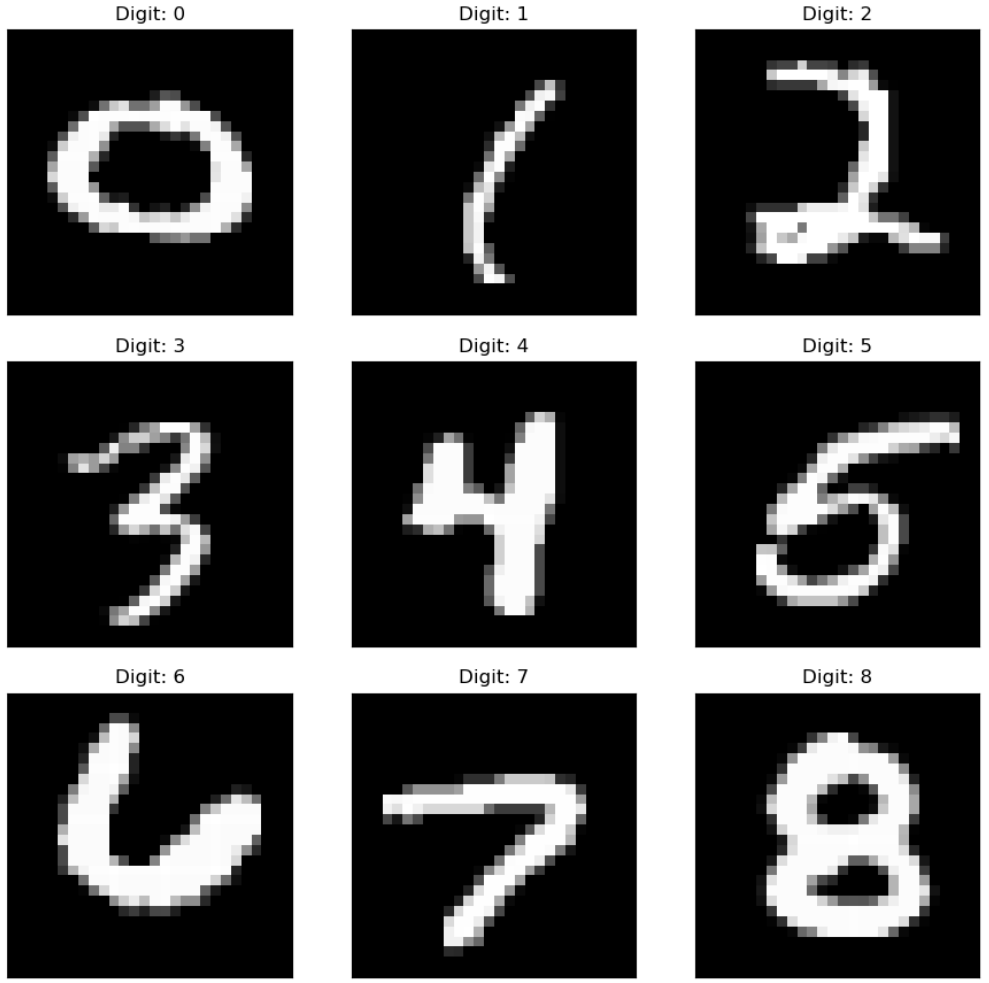
$\mathcal{D}_{test} = \{\boldsymbol{x}_m, y_m\}, M = 10{,}000$

$\boldsymbol{x}$: 28×28 grayscale images of digits 0 to 9

y : class label

Available:

`torchvision.datasets.MNIST()`



LJSpeech Dataset

$\mathcal{D}_{train} = \{\boldsymbol{x}_n, \boldsymbol{y}_n\}, N = 12{,}228$

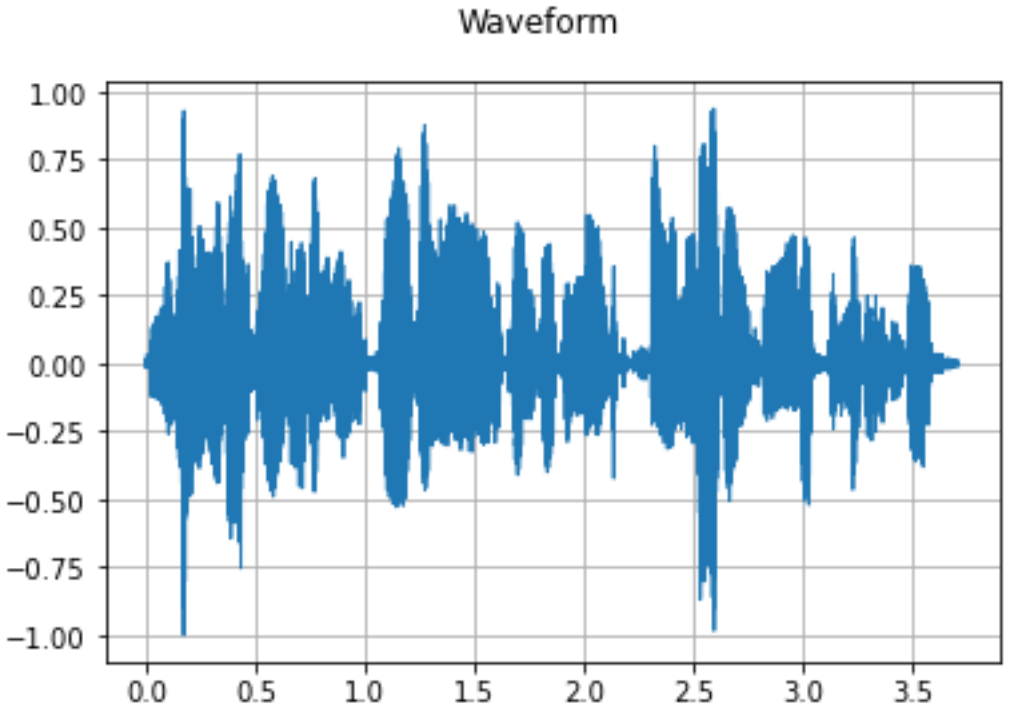
$\mathcal{D}_{test} = \{\boldsymbol{x}_m, \boldsymbol{y}_m\}, M = 523$

$\boldsymbol{x}$: text transcript

$\boldsymbol{y}$: speech

Available:

`torchaudio.datasets.LJSPEECH()`



the association was organized under the most promising auspices

Stanford Sentiment Treebank Dataset

$\mathcal{D}_{train} = \{\boldsymbol{x}_n, y_n\}, N = 67{,}349$

$\mathcal{D}_{dev} = \{\boldsymbol{x}_p, y_p\}, P = 872$

$\mathcal{D}_{test} = \{\boldsymbol{x}_m, y_m\}, M = 1{,}821$

$\boldsymbol{x}$: phrases

$\boldsymbol{y}$: sentiment (+ or 1.0 /- or 0.0)

Available:

`torchtext.datasets.SST2()`

$\boldsymbol{x}$: The gorgeously elaborate “The Lord of the Rings” …

y: 0.833

LAION5B: 5B Image-Text Pairs

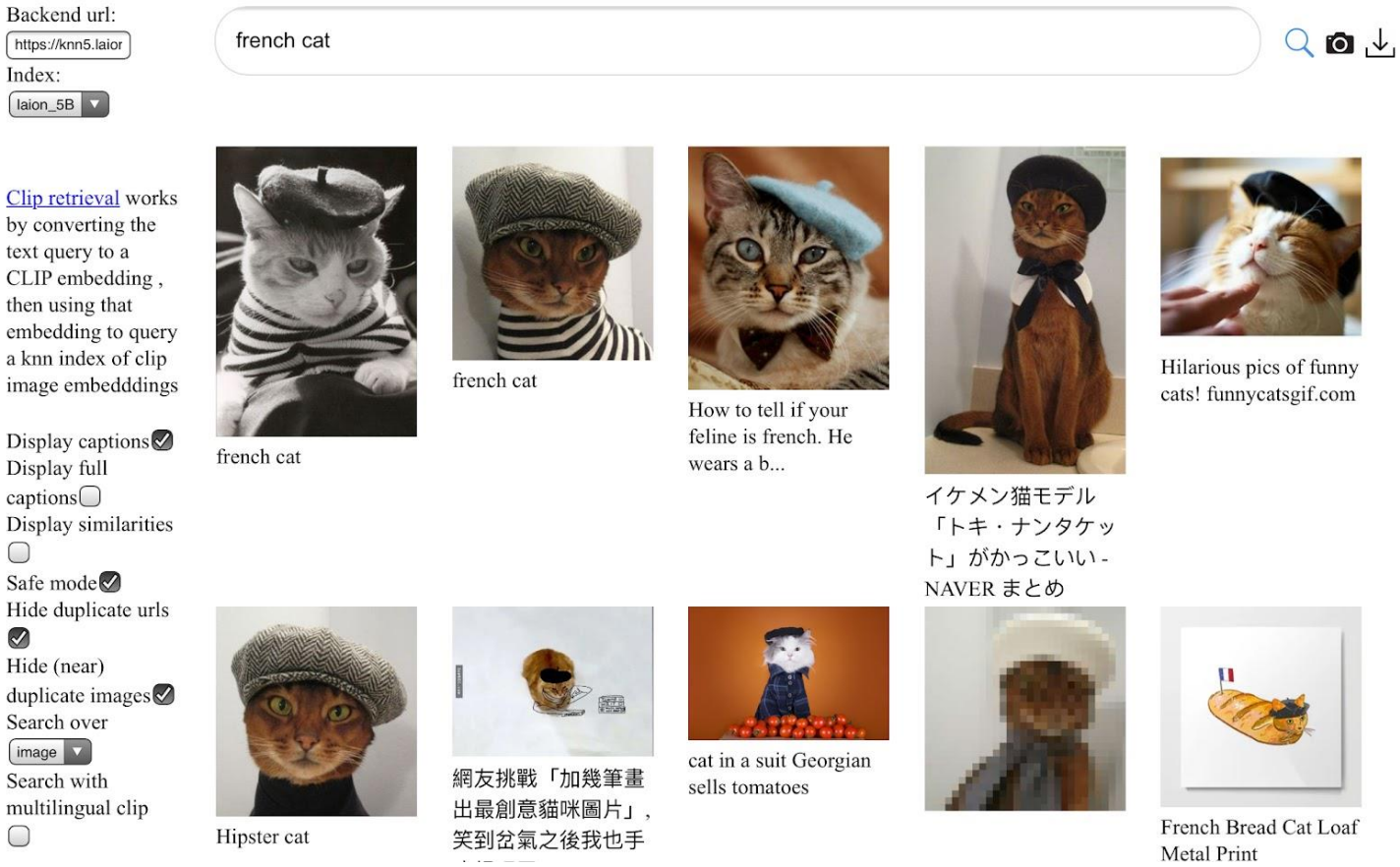
$\mathcal{D}_{train} = \{\boldsymbol{x}_n, \boldsymbol{y}_n\}, N = 5B$

$\boldsymbol{x}$: text

$\boldsymbol{y}$: image

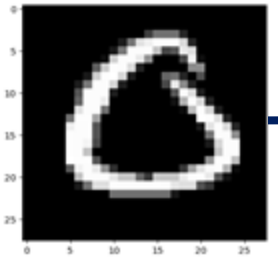
Available:

https://laion.ai/blog/laion-5b/



Task

Multi-label Classification (Recognition)



$$f(\cdot)$$

$$\begin{cases} y \in \mathbb{R} \\ y \in \mathbb{R}^{10} \end{cases} \quad \begin{matrix} y \in \{0,1,2,\dots 9\} \\ if\ one-hot \end{matrix}$$

$$\boldsymbol{x} \in \mathbb{R}^{28\times28\times1} \rightarrow \mathbb{R}^{784}$$

flatten

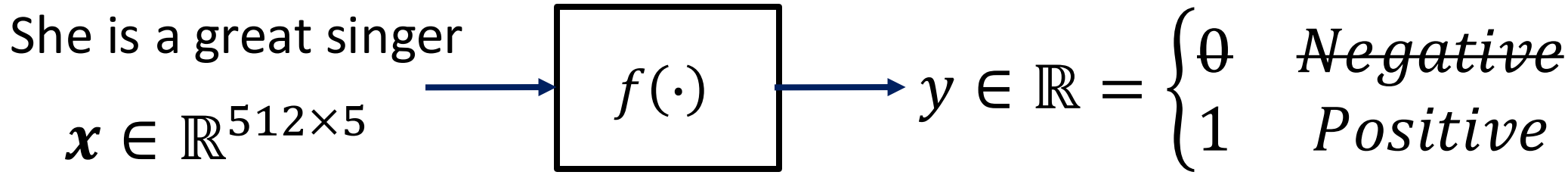
One-hot vector

$$\begin{bmatrix} 0 \\ 1 \\ 2 \\ \vdots \\ 9 \end{bmatrix} \rightarrow \begin{bmatrix} 1,0,0,0,0,0,0,0,0,0 \\ 0,1,0,0,0,0,0,0,0,0 \\ 0,0,1,0,0,0,0,0,0,0 \\ \vdots \\ 0,0,0,0,0,0,0,0,0,1 \end{bmatrix}$$

Also known as Multi-class Logistic Regression

Task: Binary Classification

Task: Sentiment Classification



Assuming embedding size is 512

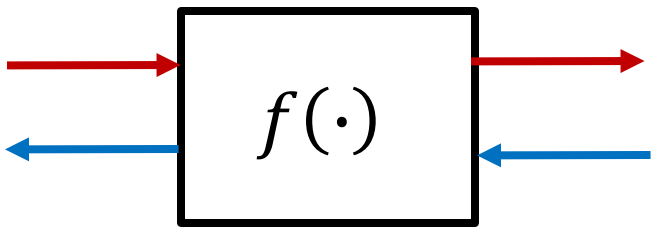
Also known as Binary Logistic Regression

Task: Sequence to Sequence

TEXT: {the association was organized under the most promising auspices}
→
PHONEME: {DH IY0 AH0 S OW2 S IY0 EY1 SH AH0 N W AH0 Z AO1 R G AH0 N AY2 Z D AH1 N D ER0 DH AH0 M OW1 S T P R AA1 M AH0 S IH0 NG AO1 S P IH0 S IH0 Z}

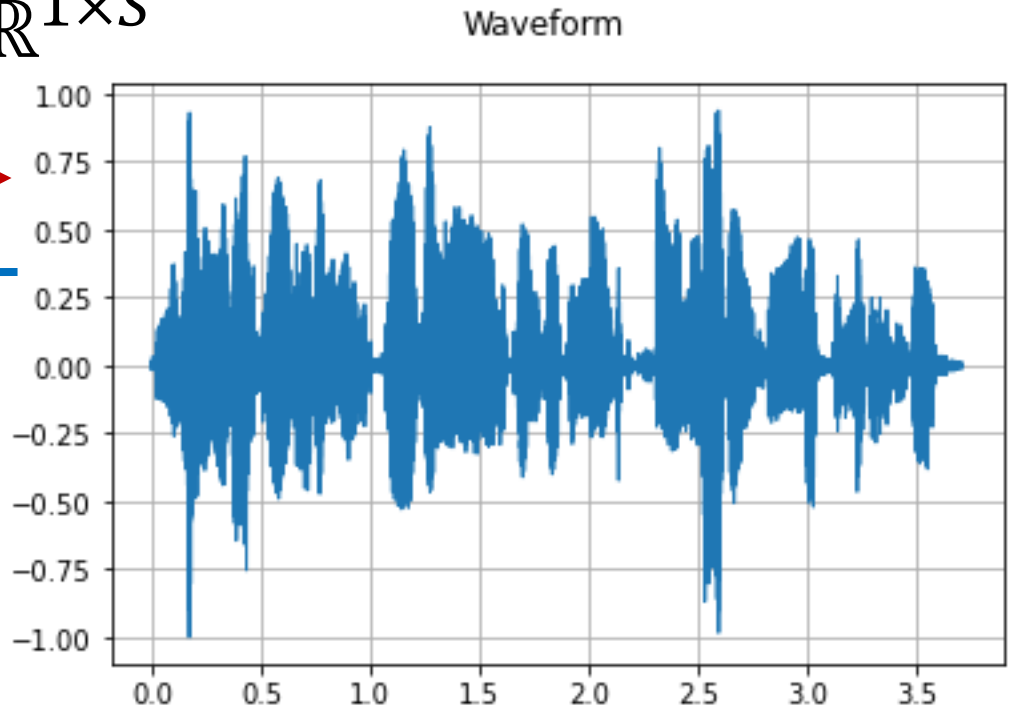
Task: Text to Speech
or Speech to Text

$\boldsymbol{x} \in \mathbb{R}^{512 \times P}$ $\boldsymbol{y} \in \mathbb{R}^{1 \times S}$



P: *Phoneme Length*
S: *#Speech Samples*

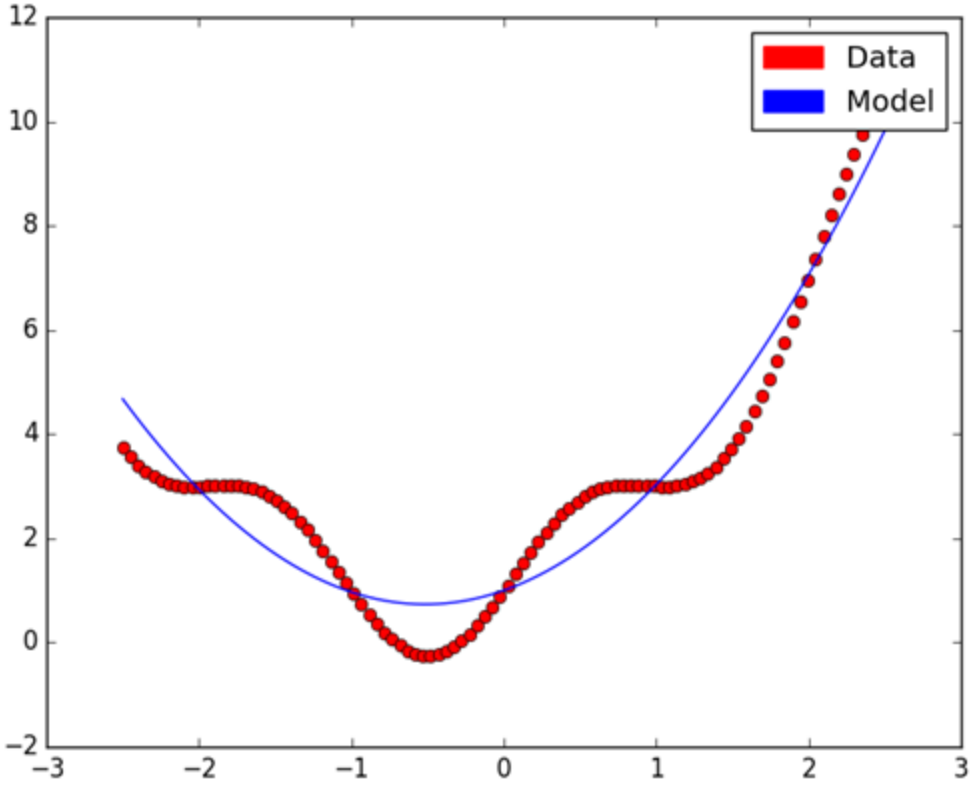
Assuming embedding size is 512



Task: Curve Fitting

$\mathcal{D}_{train} = \{x_n, y_n\}$
$\mathcal{D}_{test} = \{x_m, y_m\}$

Given: Data points



Also known as Linear Regression

Task: Image Synthesis

Flux1
@koba_1975

Stable Diffusion
@hardmaru

Task: Assistant

Input: Image and user query
by LLaVA project

```
pg
[2023-07-29 18:32:19,906] [INFO] [real_accelerator.py:110:get_accelerator] Setting ds_acc
elerator to cuda (auto detect)
Loading checkpoint shards: 100%|                    | 3/3 [00:22<00:00,  7.38s/it]
USER:
```

Performance

Accuracy (Classification) ↑

Accuracy: Classification **Classification Performance Score on** $\mathcal{D}_{test} = \{(\boldsymbol{x}_m, y_m)\}$

$$Accuracy = \frac{\#\ Correct\ Prediction}{Size\ of\ Test\ Dataset} = \frac{\sum_{m=1}^{M}(\hat{y}_m\ equals\ y_m)}{M}$$

Word Error Rate (WER) in ASR

$$WER = \frac{Substitution + Insertion + Deletion}{Number\ of\ Spoken\ Words}$$

Ground Truth: **It is the shining armor. Clear as night and day.**

$Substitution$: Replaced word(s) (e.g. *night* by *knight*)

$Insertion$: Added word(s) that is not there (e.g. instead of *the shining armor*, model transcribed it as *the knight and shining armor*)

$Deletion$: Omitted word(s) (e.g. instead of *clear as night and day*, model transcribed it as *clear night and day*)

Generative Model (Voice/Video) ↑

Mean Opinion Score (MOS) is a numerical measure of the human-judged overall quality of an event or experience.

5 Excellent

4 Good

3 Fair

2 Poor

1 Bad

Somewhere around 4.3 - 4.5 is considered an excellent quality target. Video quality becomes unacceptable below a MOS of roughly 3.5.

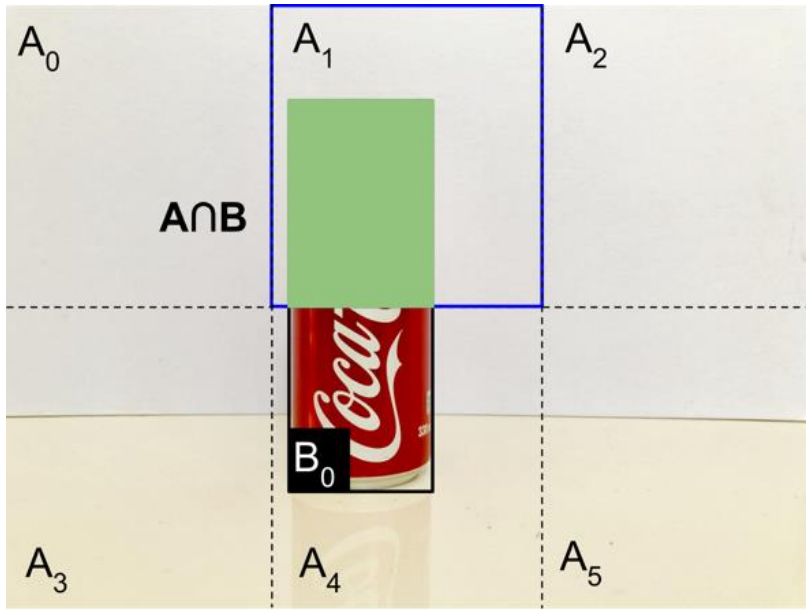
https://www.twilio.com/

Object Detection ↑

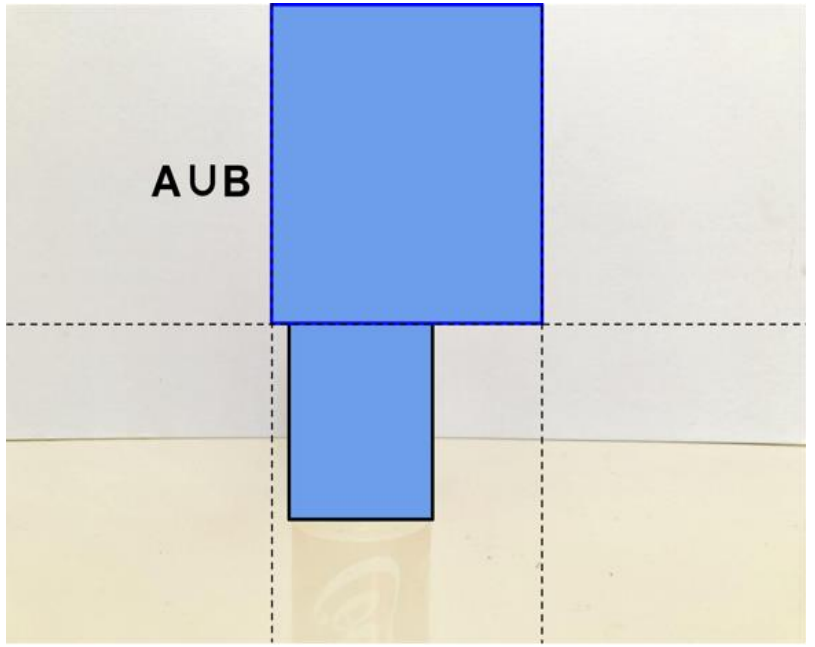
Intersection over Union (IoU)

IoU is also known as *Jaccard index:* $IoU = \frac{A \cap B}{A \cup B}$

Intersection : $A_1 \cap B_0$



Union : $A_1 \cup B_0$



Object Detection & Classification ↑

Precision (Bad guys out): ↑

$$precision = \frac{tp}{tp + fp} = \frac{True\_Positive}{True\_Positive + False\_Positive}$$

Bad guys in

Recall (Good guys in): ↑

$$recall = \frac{tp}{tp + fn} = \frac{True\_Positive}{True\_Positive + False\_Negative}$$

Good guys out

Related Performance Measures: Average Precision, F1 Score

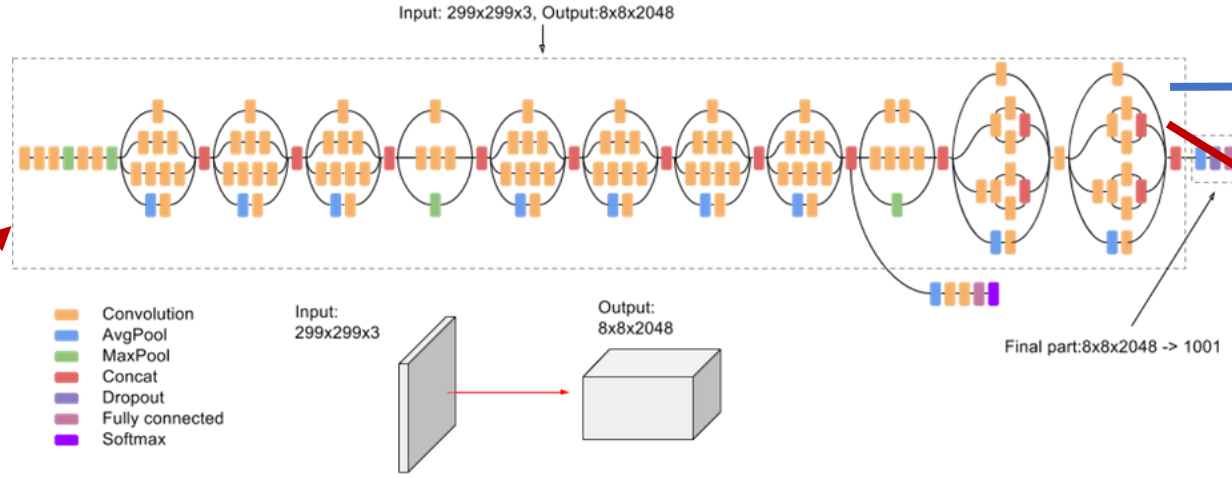
Frechet Inception Distance - d

$$d^2 = \|\boldsymbol{\mu}_r - \boldsymbol{\mu}_g\|_2^2 + Tr\left(\boldsymbol{C}_r + \boldsymbol{C}_r - 2\sqrt{\boldsymbol{C}_r\boldsymbol{C}_g}\right)$$

Feature vectors Gaussian mean and covariance of real and generated images: $(\boldsymbol{\mu}_r, \boldsymbol{C}_r), (\boldsymbol{\mu}_g, \boldsymbol{C}_g)$

Real

Generated



$(\boldsymbol{\mu}_r, \boldsymbol{C}_r)$

$(\boldsymbol{\mu}_g, \boldsymbol{C}_g)$

Pre-trained InceptionV3 Model

Housel etal, GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, NIPS 2017

Evaluating Assistants

- Multimodal QA: GPT4 evaluates the helpfulness, relevance, accuracy, and level of details of the responses from the assistants, and give an overall score on a scale of 1 to 10
- ScienceQA: Accuracy

Learning

Learning by Minimizing a Loss Function

Ground truth: y_n

Prediction: $\hat{y}_n$

Loss Function: $\mathcal{L} = l(y_n, \hat{y}_n)$

Machine Learning: Use dataset, $\mathcal{D}_{train} = \{(\boldsymbol{x}_n, y_n)\}, \quad n = 1 \ldots N$ to estimate the parameters $\boldsymbol{\theta}$ by minimizing $L = l(y_n, \hat{y}_n)$ using an optimizer

Assumption: $(\boldsymbol{x}_1, y_1) \ldots (\boldsymbol{x}_n, y_n)$ are IID

IID: Independent and identically distributed

Empirical Risk Minimization (ERM)

$$R_{emp}(f, \boldsymbol{X}, \boldsymbol{y}) = \frac{1}{N} \sum_{i=1}^{n} l(y_n, \hat{y}_n)$$

Where $\boldsymbol{X} := [x_1, \ldots, x_n]^T \in \mathbb{R}^{N \times D}$ and $\boldsymbol{y} := [y_1, \ldots, y_n]^T \in \mathbb{R}^N$

Least Squares Loss

$$\mathcal{L} = l(y_n, \hat{y}_n) = (y_n - \hat{y}_n)^2$$

$$R_{emp}(f, \boldsymbol{X}, \boldsymbol{y}) = \frac{1}{N}\sum_{i=1}^{n}(y_n - \hat{y}_n)^2 = \frac{1}{N}\sum_{i=1}^{n}\left(y_n - f(\boldsymbol{x}_n, \boldsymbol{\theta})\right)^2$$

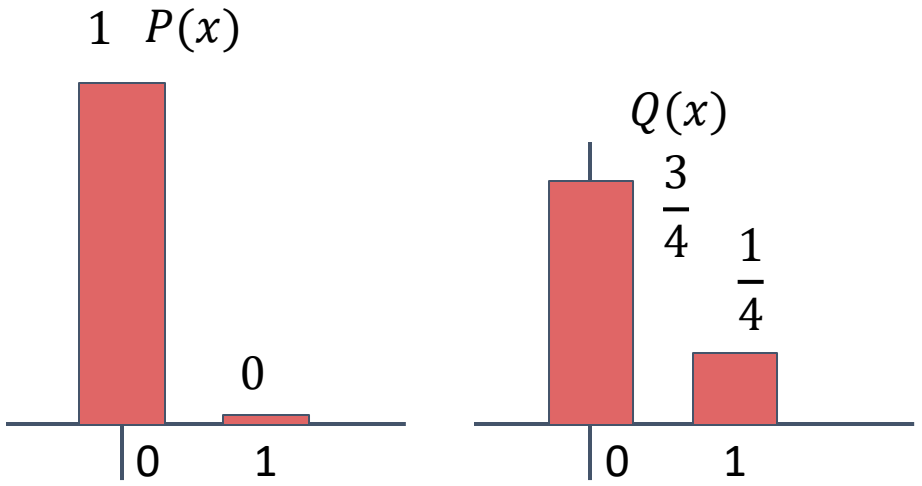
Loss Functions

Loss Function	Equation
Mean Squared Error (MSE)	$\sum_{i=1}^{categories} \left(y_i^{label} - y_i^{prediction}\right)^2$
Mean Absolute Error (MAE)	$\sum_{i=1}^{categories} \left\lvert y_i^{label} - y_i^{prediction} \right\rvert$
Categorical Cross Entropy (CE)	$-\sum_{i=1}^{categories} y_i^{label} \log y_i^{prediction}$
Binary Cross Entropy (BCE)	$-y_1^{label} \log y_1^{prediction} - \left(1 - y_1^{label}\right) \log\left(1 - y_1^{prediction}\right)$

Categorical Cross-Entropy (CE)

For discrete distribution, Categorical Cross-Entropy is:

$$CE = H(P, Q) = -\underbrace{\sum_i P(x_i)}_{\text{Empirical Label}} \underbrace{\log Q(x_i)}_{\text{Predicted Label}}$$



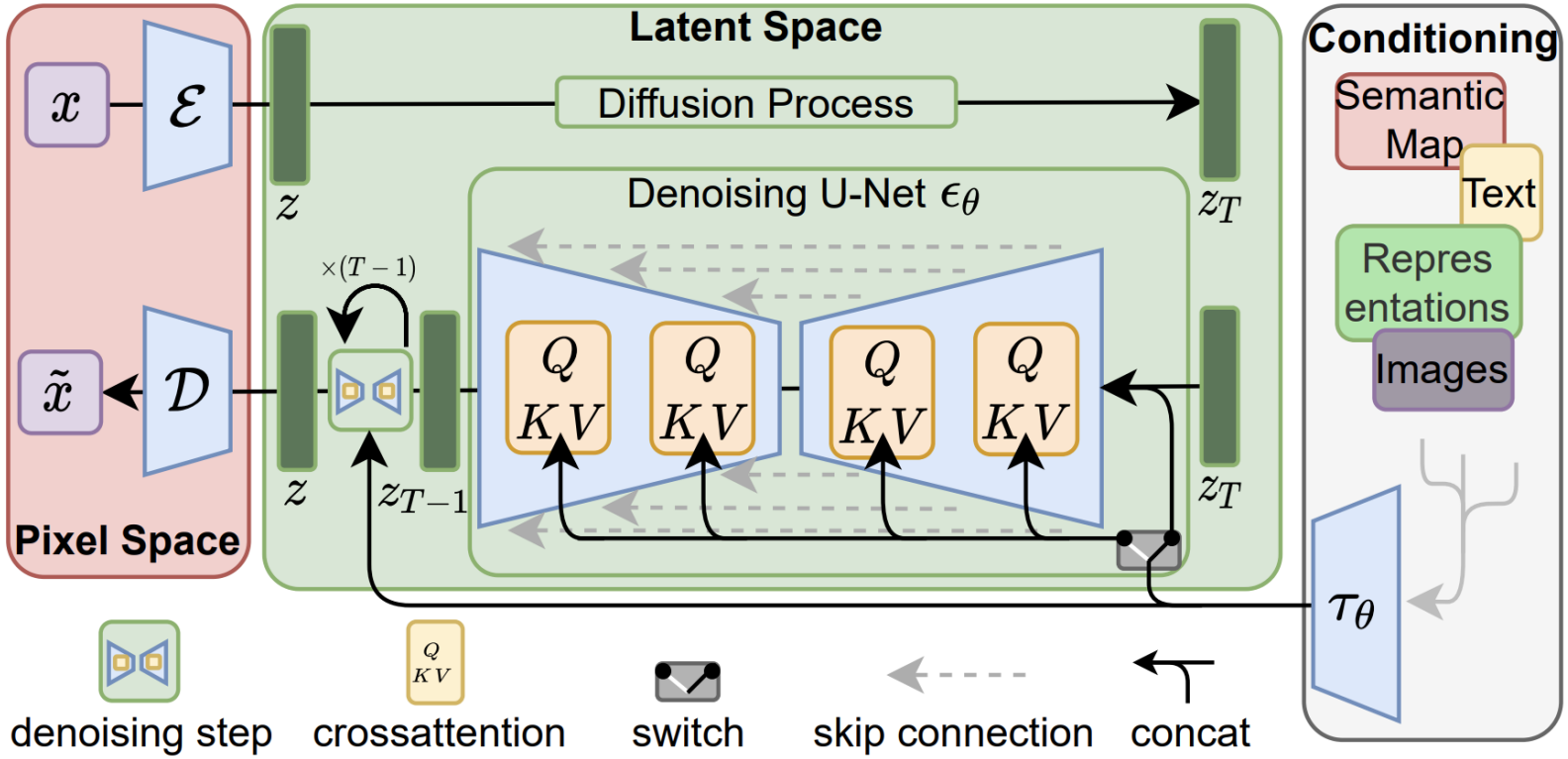
What is CE?

$$CE = -\left(1 \log \frac{3}{4} + 0 \log \frac{1}{4}\right) = 0.29$$

Minimizing $H(P, Q)$ minimizes the distance of prediction model Q from the empirical model P

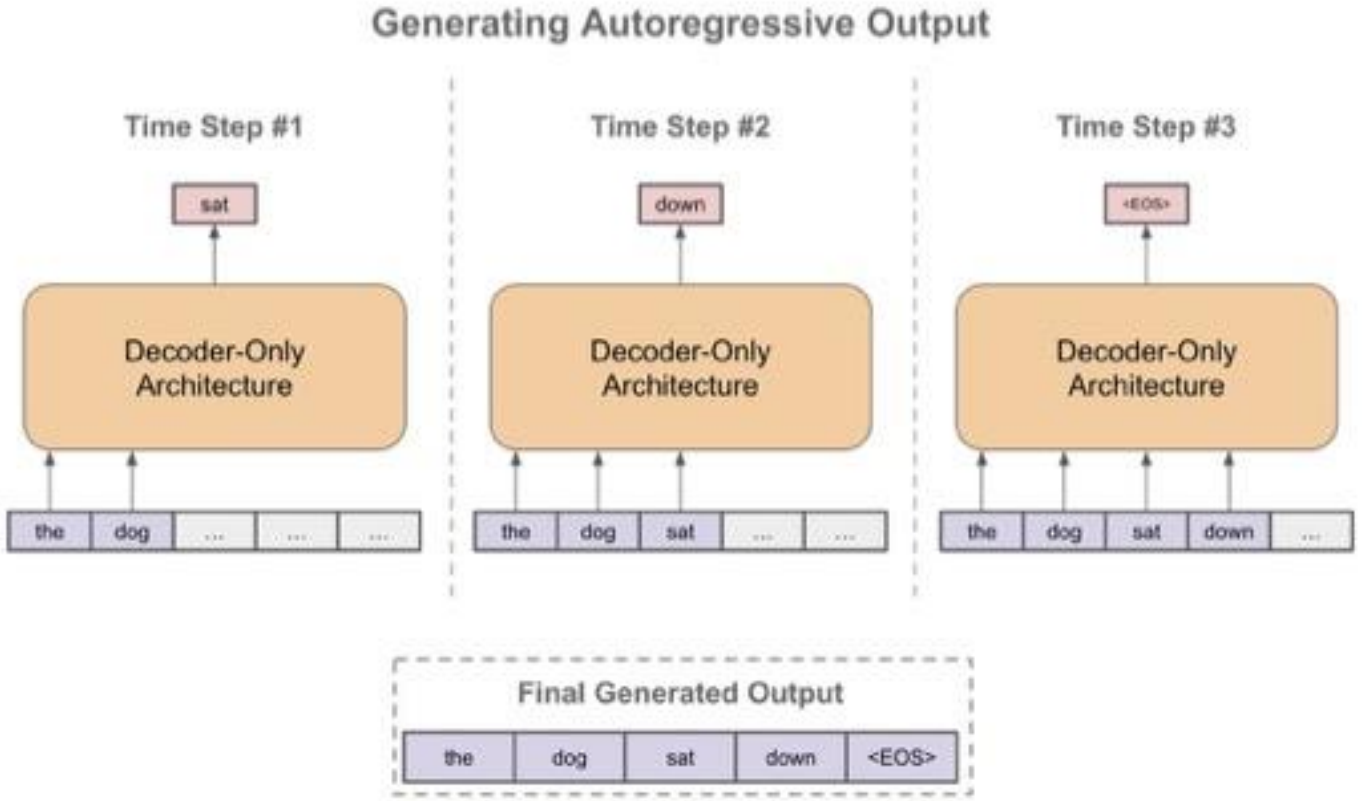
Latent Diffusion Model (LDM) Loss

$$L_{LDM} := \mathbb{E}_{\mathcal{E}(x),y,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t, \tau_\theta(y))\|_2^2\right]$$



Rombach et al, High-Resolution Image Synthesis with Latent Diffusion Models, CVPR2021

GPT CE Loss



GPT2: Radford, Alec, et al. "Language models are unsupervised multitask learners." *OpenAI blog* 1.8 (2019): 9.

GPT3: Brown, Tom, et al. "Language models are few-shot learners." Advances in neural information processing systems 33 (2020): 1877-1901.

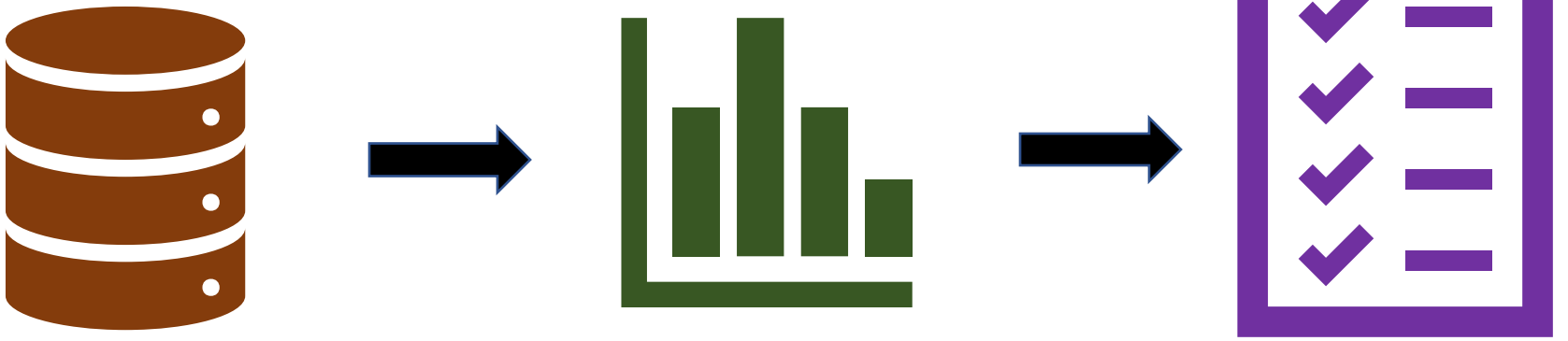
Machine Learning Pipeline

$\mathcal{D}_{train}$

$f(\boldsymbol{x}, \boldsymbol{\theta}^0) \rightarrow f(\boldsymbol{x}, \boldsymbol{\theta}^*)$

Minimize $\mathcal{L}$

$\mathcal{D}_{test}$



Experience

Task

Performance

End