DSC291: Machine Learning with Few Labels

Self-supervised Learning

Zhiting Hu

Lecture 8, April 24, 2025

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

Outline

- Self-Supervised Learning (SSL)
  - Contrastive learning

- Paper presentation:
  - Kaiming Tao, Wenqi Li: “Transformers without Normalization”

“X”-supervised learning

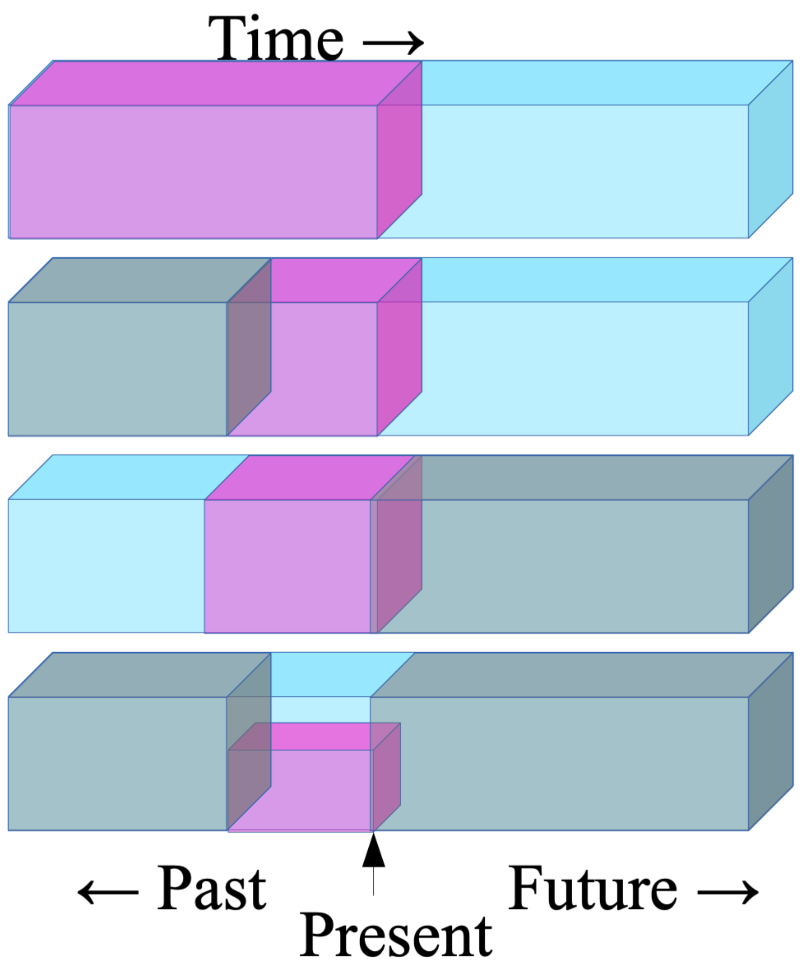
- Supervised learning
- Unsupervised learning
- Self-supervised learning
- Weakly-/distantly-supervised learning
- Semi-supervised learning
- …

Self-Supervised Learning

- Given an observed data instance $\boldsymbol{t}$
- One could derive various supervision signals based on the structure of the data
- By applying a "split" function that artificially partition $\boldsymbol{t}$ into two parts
  - $(\boldsymbol{x}, \boldsymbol{y}) = split(\boldsymbol{t})$
  - sometimes split in a stochastic way
- Treat $\boldsymbol{x}$ as the input and $\boldsymbol{y}$ as the output
- Train a model $p_\theta(\boldsymbol{y}|\boldsymbol{x})$

Self-Supervised Learning: Examples

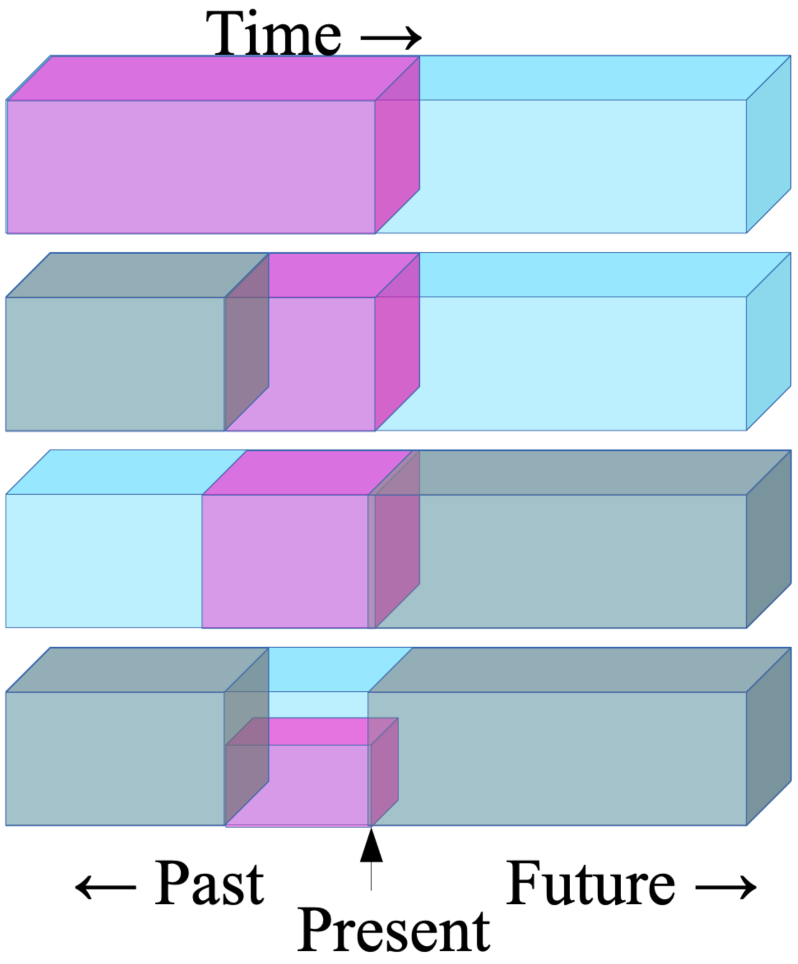
- Predict any part of the input from any other part.
- Predict the future from the past.
- Predict the future from the recent past.
- Predict the past from the present.
- Predict the top from the bottom.



[Courtesy: Lecun “Self-supervised Learning”]

Self-Supervised Learning: Examples

- Predict any part of the input from any other part.
- Predict the future from the past.
- Predict the future from the recent past.
- Predict the past from the present.
- Predict the top from the bottom.
- Predict the occluded from the visible
- Pretend there is a part of the input you don't know and predict that.



Self-Supervised Learning: Motivation (I)

- **Our brains do this all the time**

- **Filling in the visual field at the retinal blind spot**
- **Filling in occluded images, missing segments in speech**
- **Predicting the state of the world from partial (textual) descriptions**
- **Predicting the consequences of our actions**
- **Predicting the sequence of actions leading to a result**

- **Predicting any part of the past, present or future percepts from whatever information is available.**

[Courtesy: Lecun "Self-supervised Learning"]

Self-Supervised Learning: Motivation (I)

- Successfully learning to predict everything from everything else would result in the accumulation of lots of background knowledge about how the world works

- The model is forced to learn what we really care about, e.g. a semantic representation, in order to solve the prediction problem

[Courtesy: Lecun “Self-supervised Learning”]

[Courtesy: Zisserman “Self-supervised Learning”]

Self-Supervised Learning: Motivation (II)

- The machine predicts any part of its input from any observed part
  - **A lot of** supervision signals in each data instance

- Untapped/availability of vast numbers of unlabeled text/images/videos..
  - Facebook: one billion images uploaded per day
  - 300 hours of video are uploaded to YouTube every minute

[Courtesy: Zisserman “Self-supervised Learning”]

Self-Supervised Learning (SSL): Examples

- SSL from text
- SSL from images
- SSL from videos

Self-Supervised Learning from Text

Examples:

- Language models
- Learning contextual text representations

Language Models

- Calculates the probability of a sentence:
  - Sentence:

$$\boldsymbol{y} = (y_1, y_2, \dots, y_T)$$

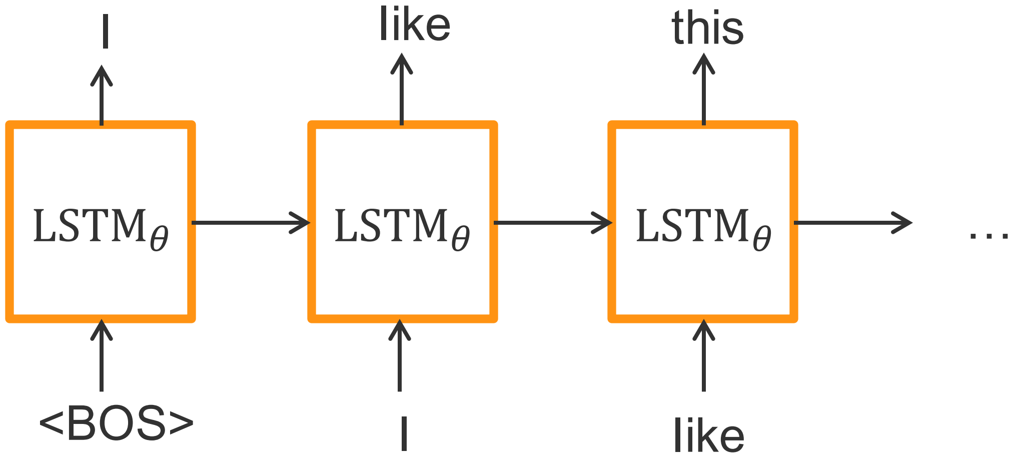
$$p_\theta(\boldsymbol{y}) = \prod_{t=1}^{T} p_\theta(y_t \mid \boldsymbol{y}_{1:t-1})$$

Example:

(I, like, this, …)

$$\cdots p_\theta\,(like \mid I)\, p_\theta(this \mid I, like) \cdots$$

Model: LSTM RNN



Language Models

- Calculates the probability of a sentence:
  - Sentence:

Example:

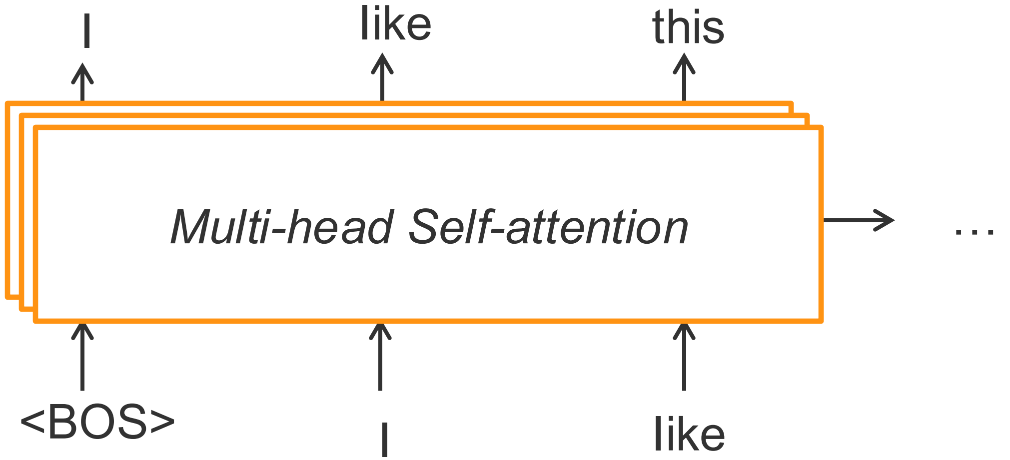
$$\boldsymbol{y} = (y_1, y_2, \dots, y_T)$$

(I, like, this, …)

$$p_\theta(\boldsymbol{y}) = \prod_{t=1}^{T} p_\theta(y_t \mid \boldsymbol{y}_{1:t-1})$$

$$\cdots p_\theta\,(like \mid I)\, p_\theta(this \mid I, like) \cdots$$

Model: Transformer



Language Models: Training

- Given data example $\boldsymbol{y}^*$
- Minimizes negative log-likelihood of the data

$$\min_\theta \mathcal{L}(\theta) \quad = -\log p_\theta(\boldsymbol{y}^*) = -\prod_{t=1}^{T} p_\theta(y_t^* \mid \boldsymbol{y}_{1:t-1}^*)$$

  - Next word prediction

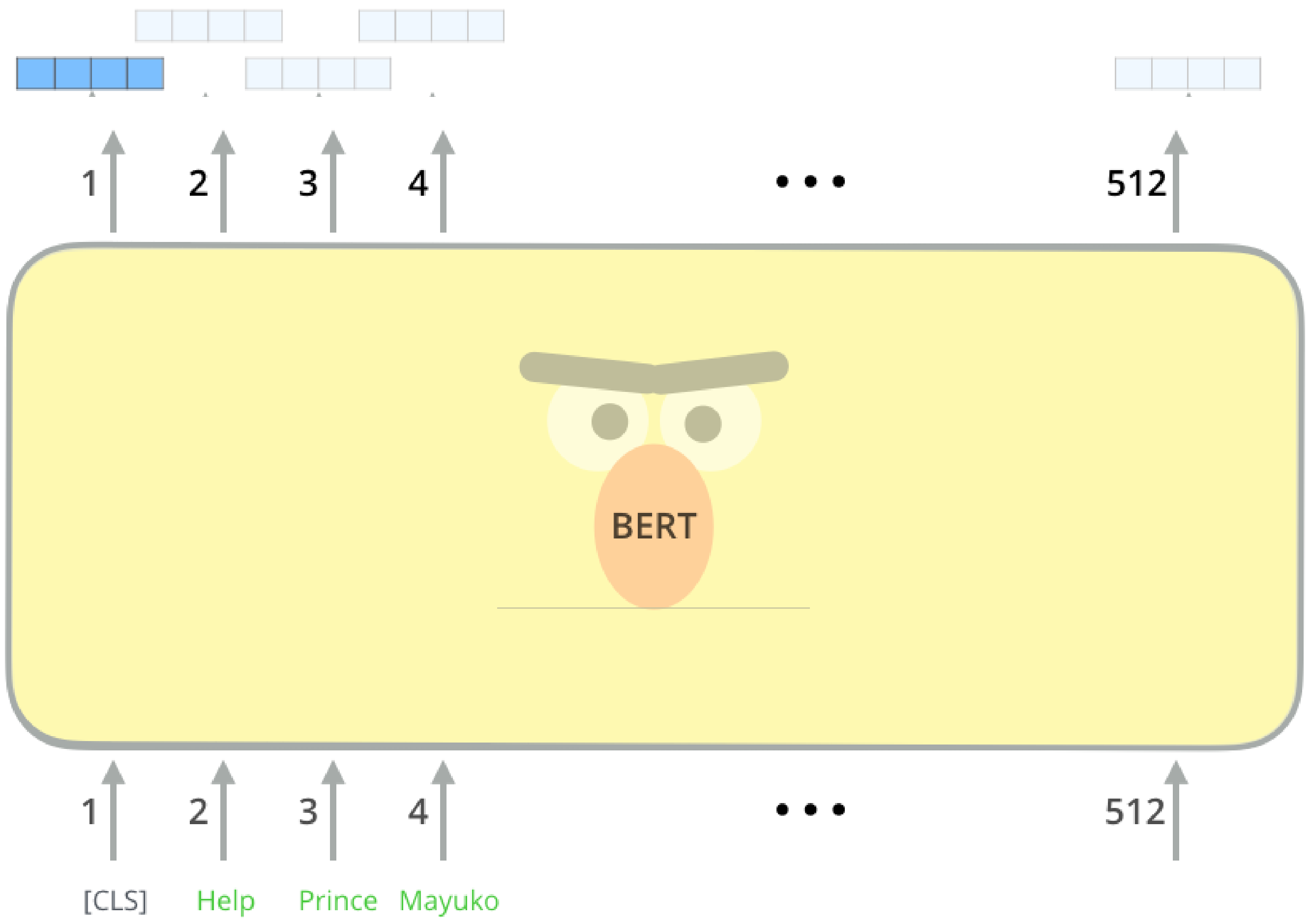
Self-Supervised Learning from Text

Examples:

- Language models
- Learning contextual text representations

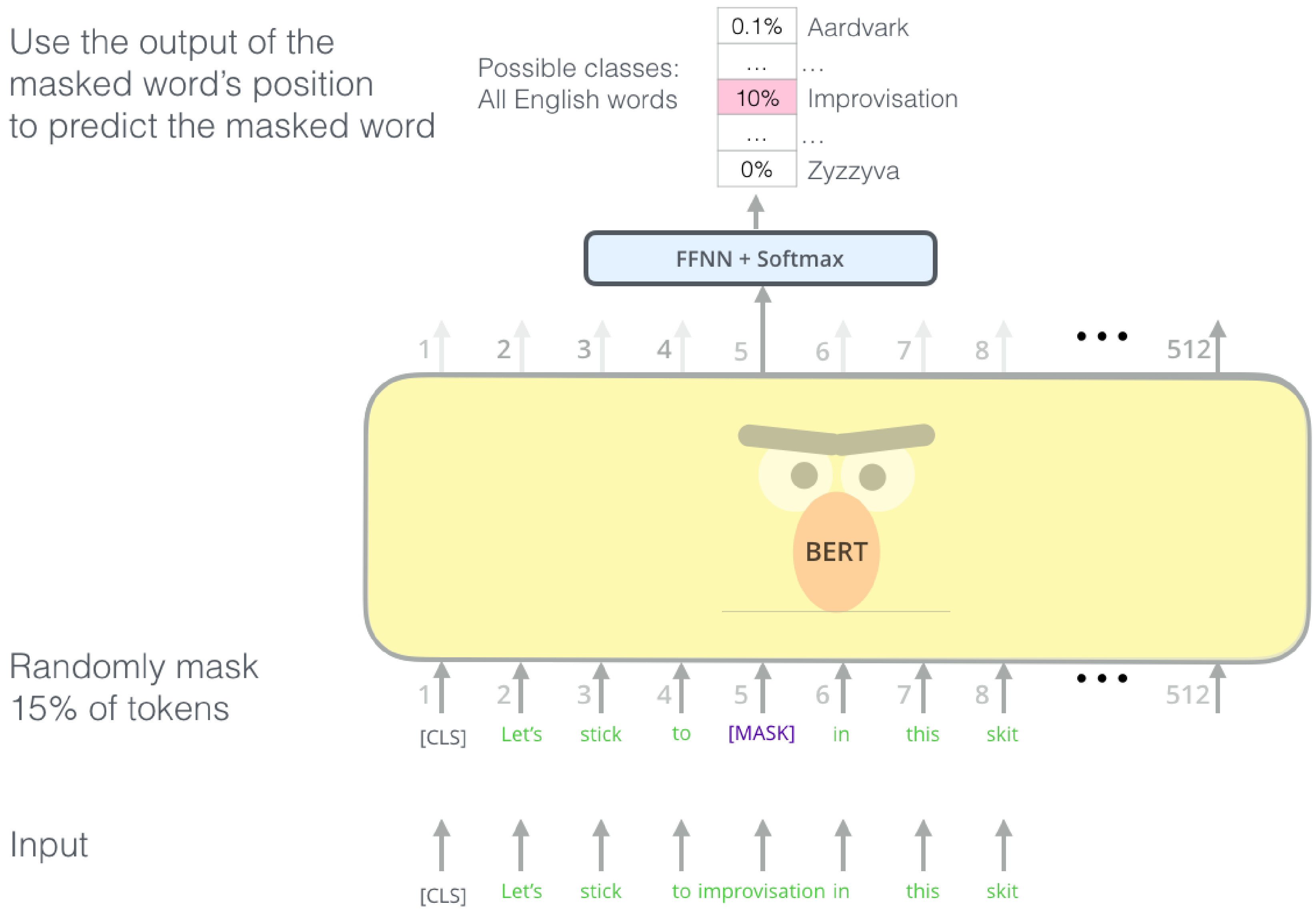
BERT

- BERT: A bidirectional model to extract contextual word embedding



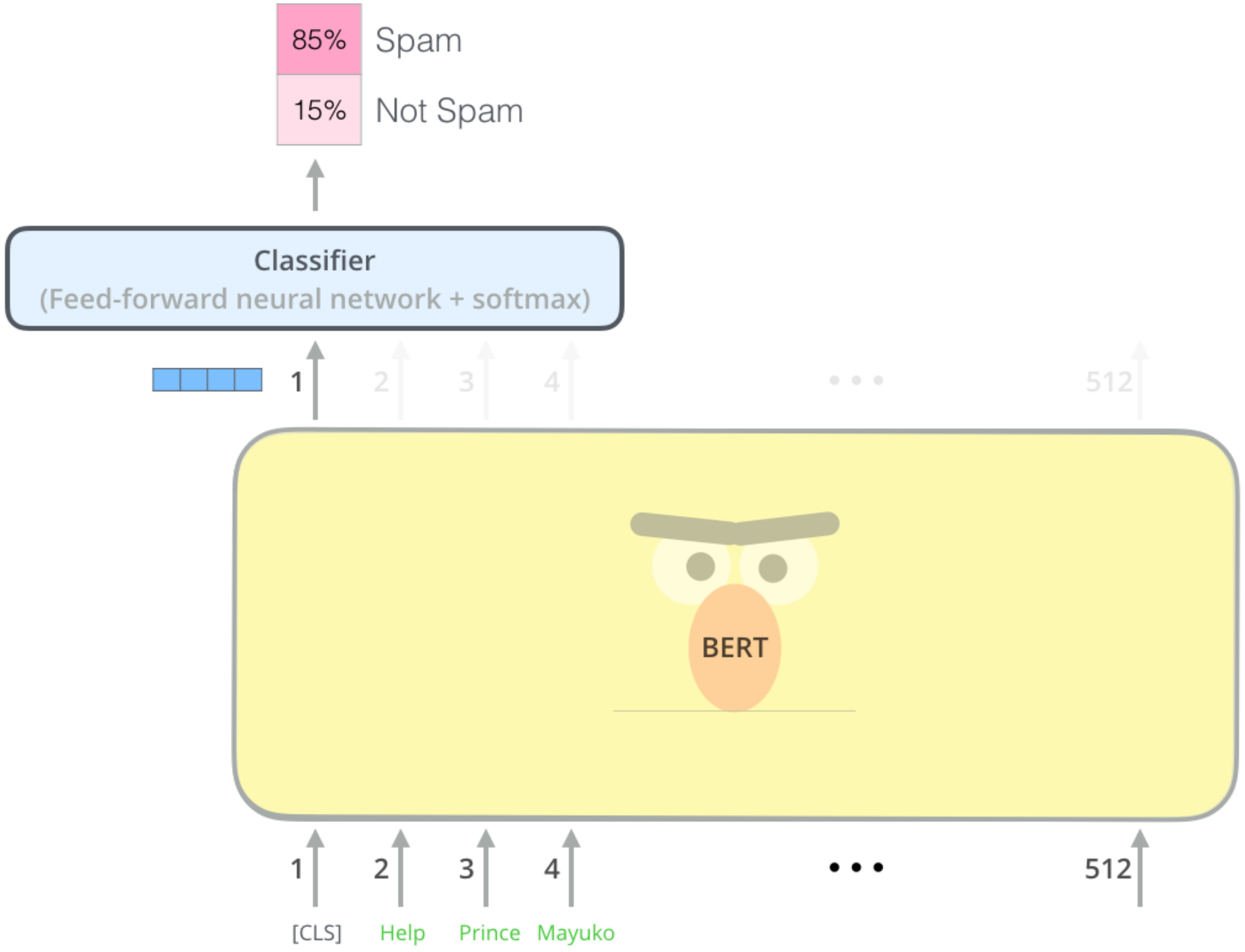
BERT: Pre-training with Self-supervised Learning

- Masked LM



BERT: Downstream Fine-tuning

- Use BERT for sentence classification



BERT Results

- Huge improvements over SOTA on 12 NLP task

System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.9	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	88.1	91.3	45.4	80.0	82.3	56.0	75.2
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.1	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**91.1**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**81.9**

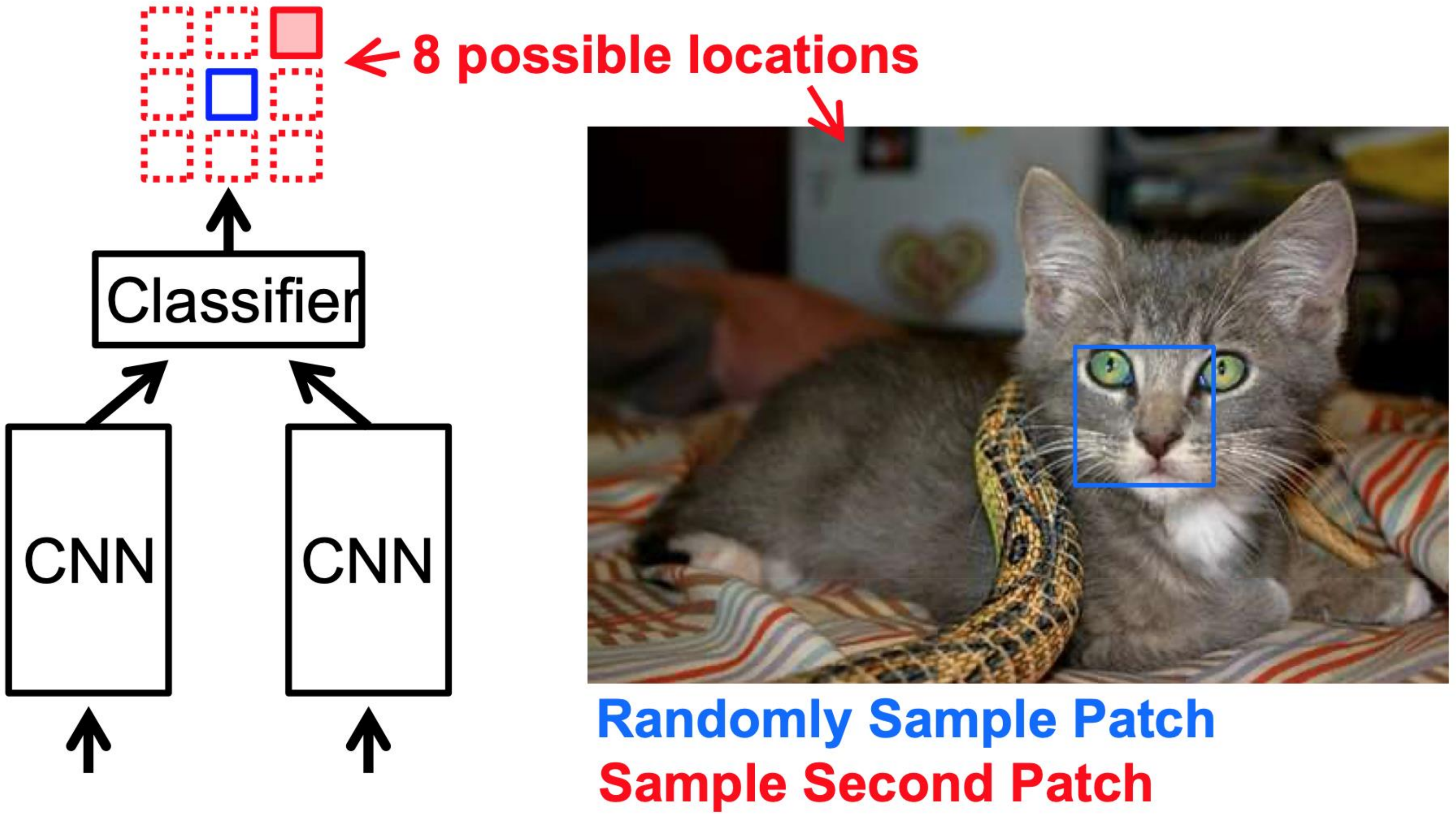
Table 1: GLUE Test results, scored by the GLUE evaluation server. The number below each task denotes the number of training examples. The "Average" column is slightly different than the official GLUE score, since we exclude the problematic WNLI set. OpenAI GPT = (L=12, H=768, A=12); $\text{BERT}_{\text{BASE}}$ = (L=12, H=768, A=12); $\text{BERT}_{\text{LARGE}}$ = (L=24, H=1024, A=16). BERT and OpenAI GPT are single-model, single task. All results obtained from https://gluebenchmark.com/leaderboard and https://blog.openai.com/language-unsupervised/.

Self-Supervised Learning (SSL): Examples

- SSL from text
- SSL from images
- SSL from videos

SSL from Images, EX (I): relative positioning

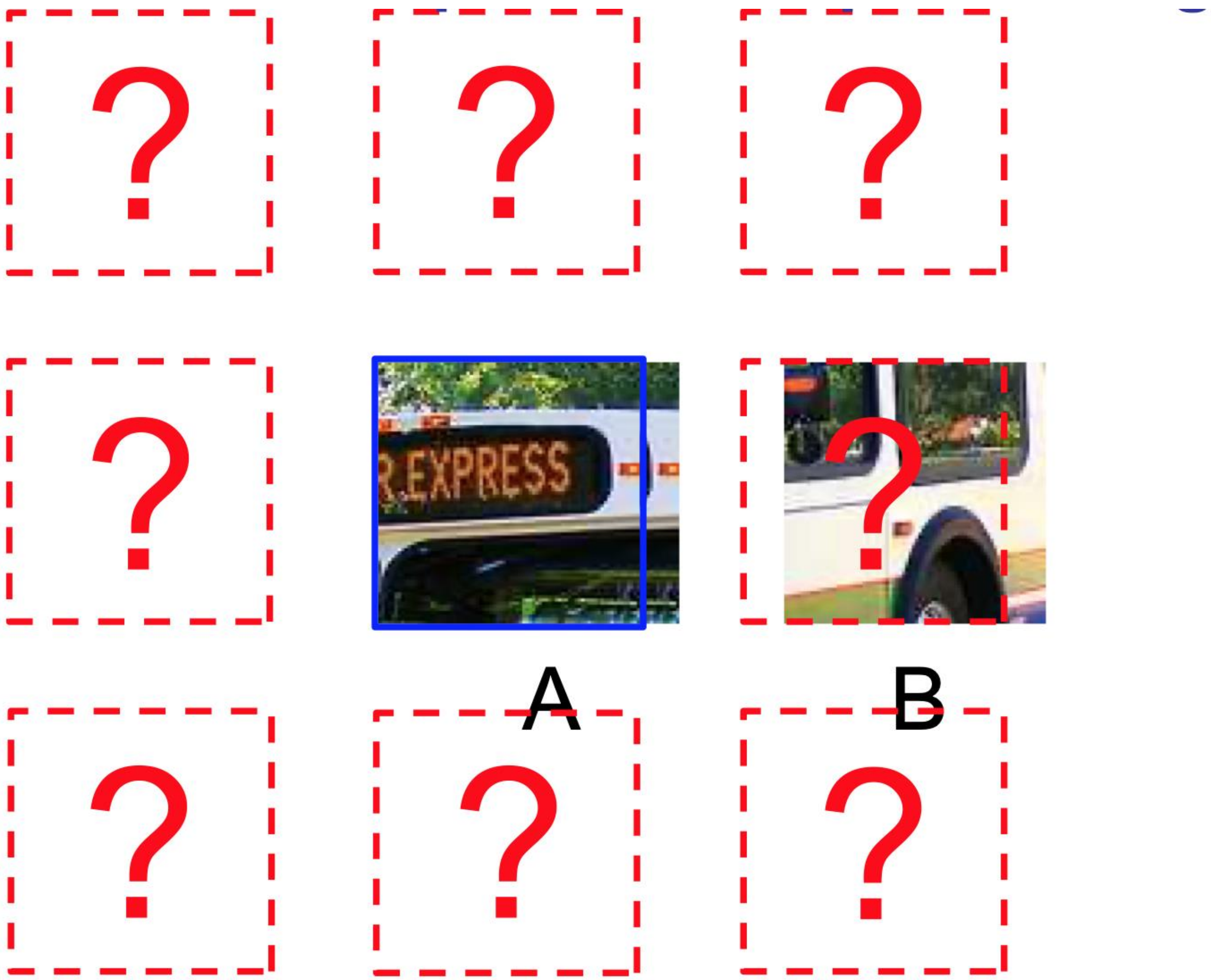
Train network to predict relative position of two regions in the same image



Unsupervised visual representation learning by context prediction, Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015

[Courtesy: Zisserman “Self-supervised Learning”]

SSL from Images, EX (I): relative positioning



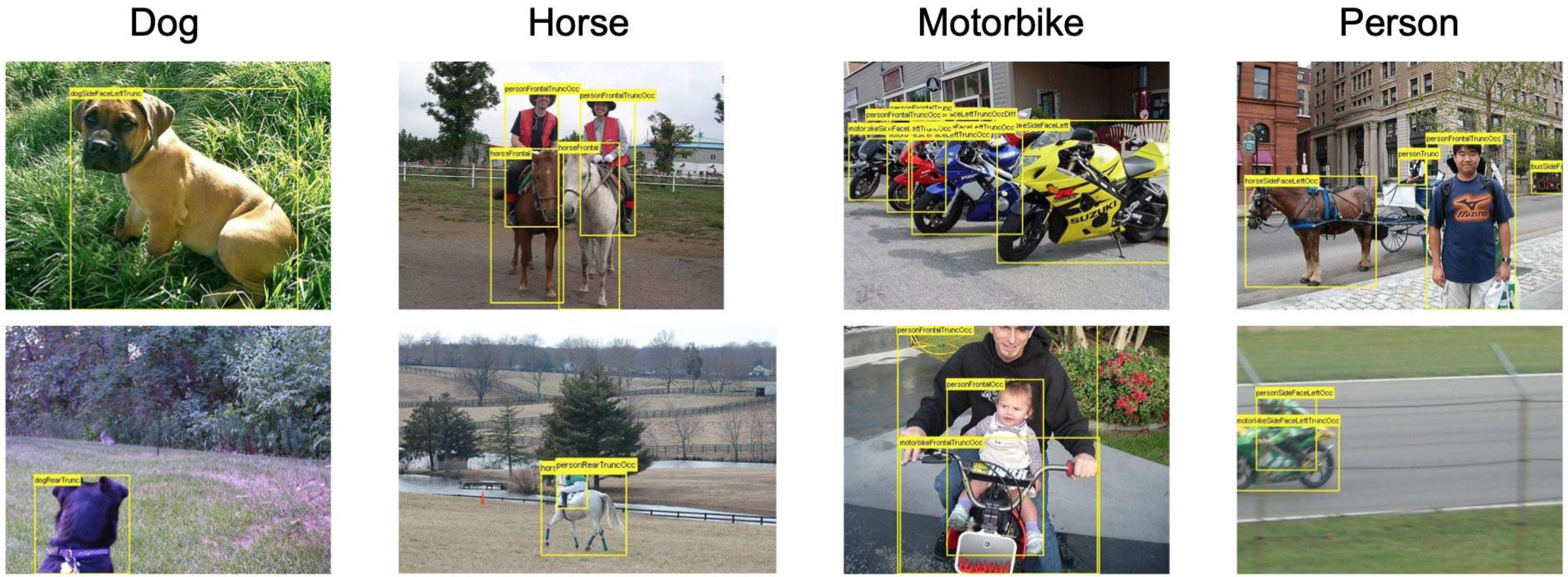
Unsupervised visual representation learning by context prediction,
Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015

[Courtesy: Zisserman "Self-supervised Learning"]

SSL from Images, EX (I): relative positioning

Evaluation: PASCAL VOC Detection

- 20 object classes (car, bicycle, person, horse …)
- Predict the bounding boxes of all objects of a given class in an image (if any)



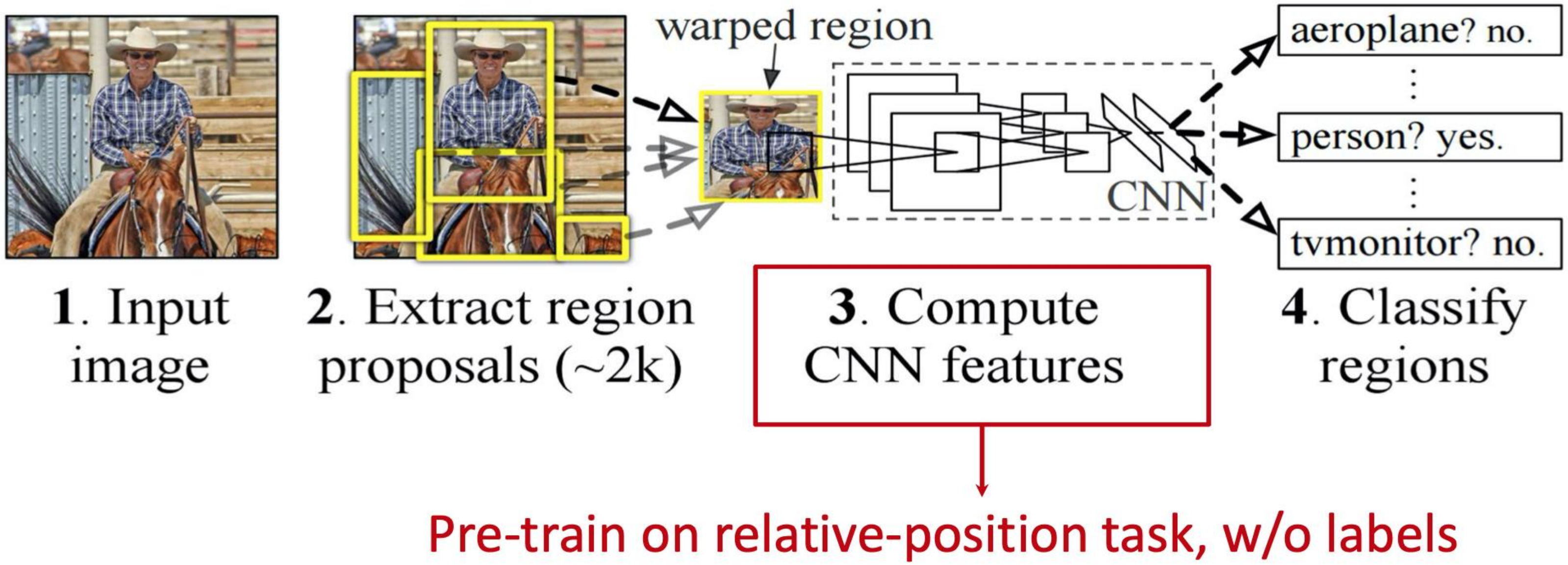
[Courtesy: Zisserman “Self-supervised Learning”]

SSL from Images, EX (I): relative positioning

Evaluation: PASCAL VOC Detection

- Pre-train CNN using self-supervision (no labels)
- Train CNN for detection in R-CNN object category detection pipeline

R-CNN

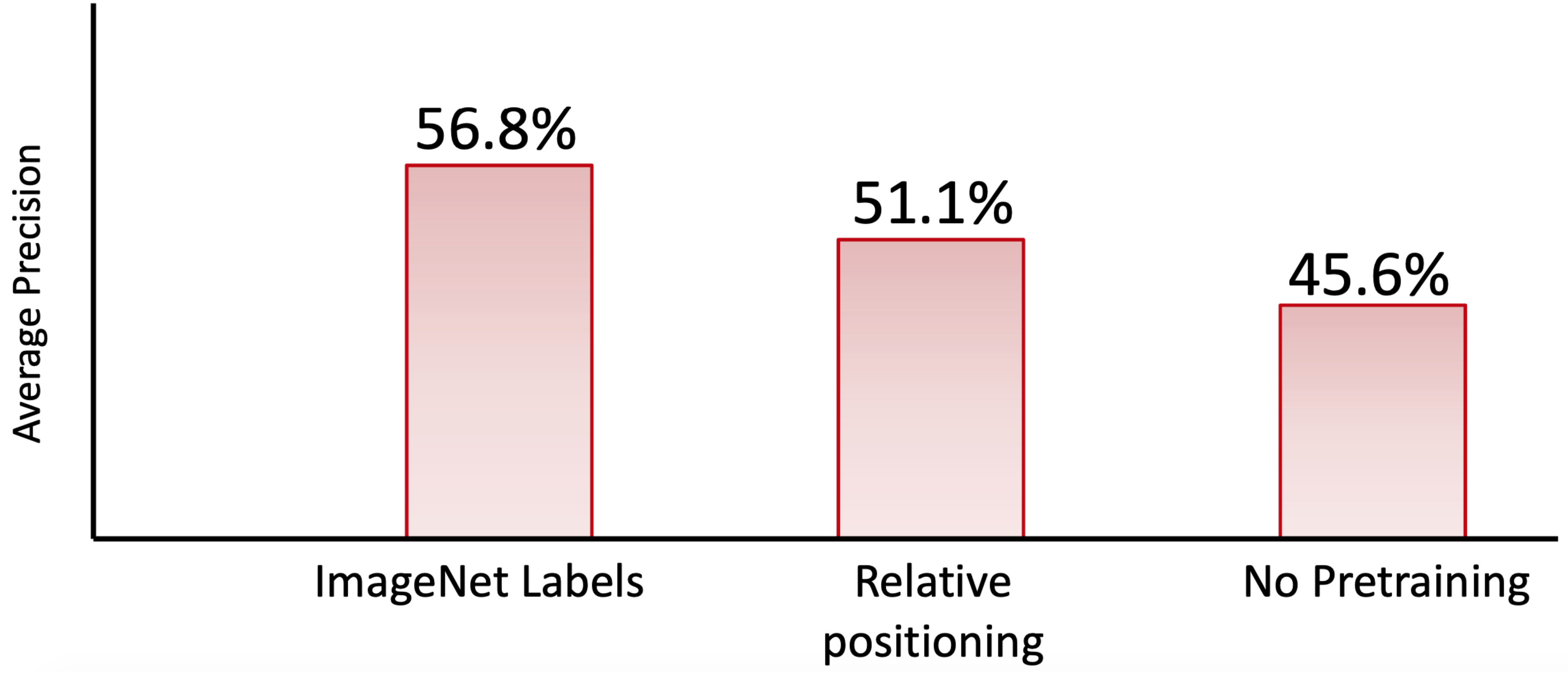


[Girshick et al. 2014]

[Courtesy: Zisserman "Self-supervised Learning"]

SSL from Images, EX (I): relative positioning

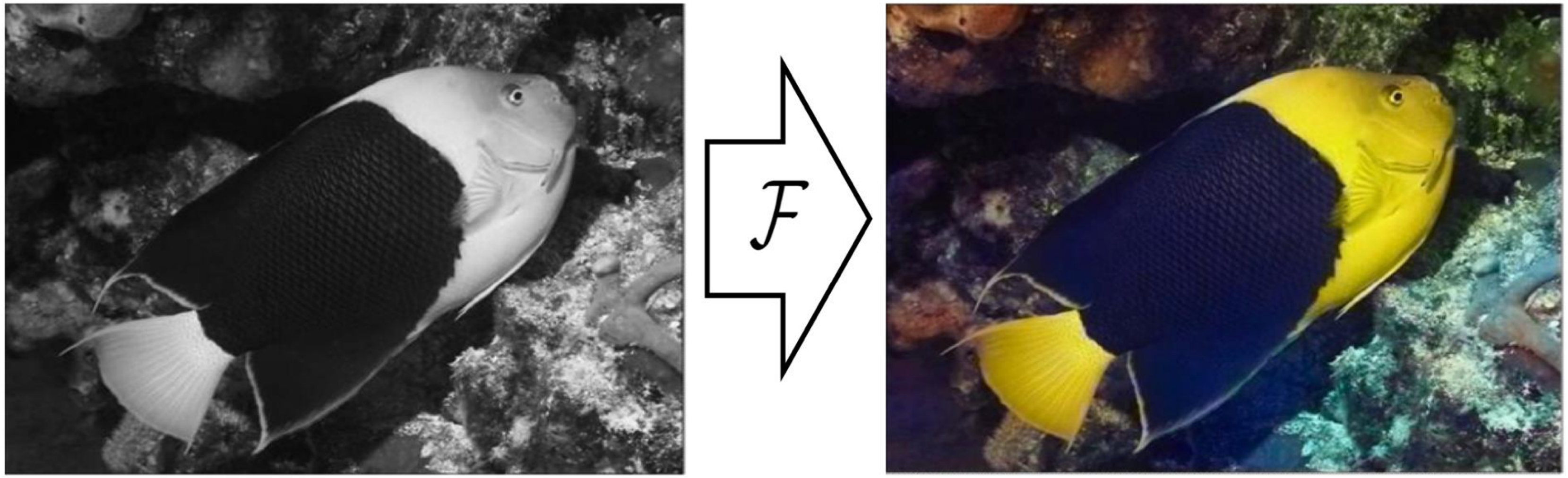
Evaluation: PASCAL VOC Detection



[Courtesy: Zisserman “Self-supervised Learning”]

SSL from Images, EX (II): colorization

Train network to predict pixel colour from a monochrome input

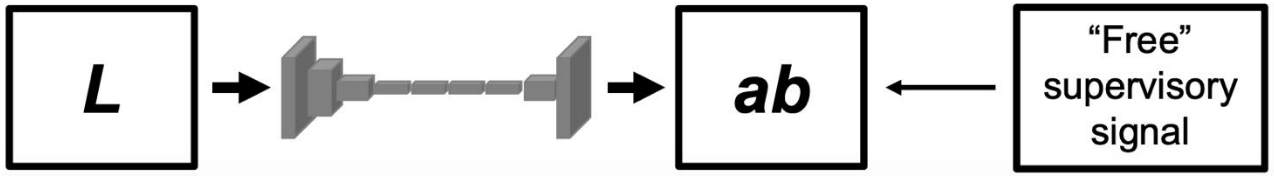


Grayscale image: *L* channel

$$\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$$

Concatenate (L,ab)

$$(\mathbf{X}, \widehat{\mathbf{Y}})$$

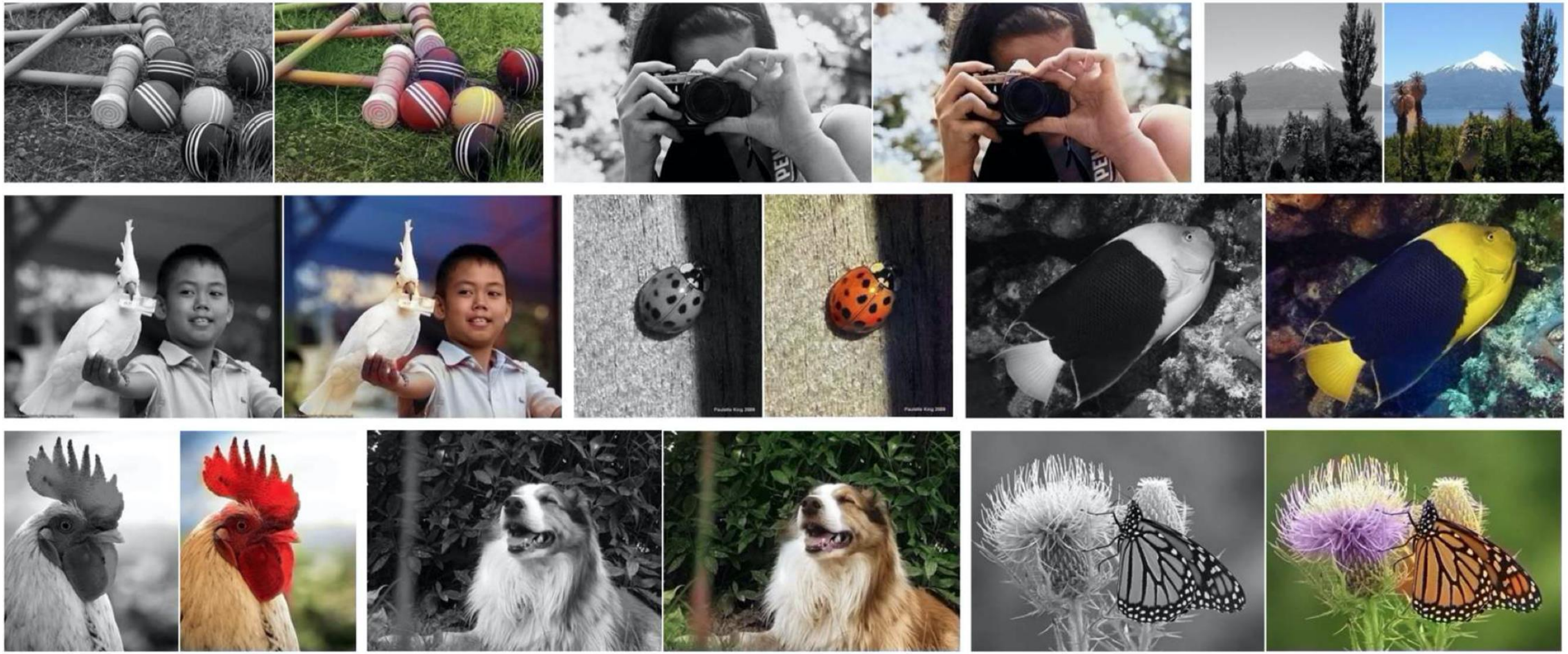


[Courtesy: Zisserman "Self-supervised Learning"]

Colorful Image Colorization, Zhang et al., ECCV 2016

SSL from Images, EX (II): colorization

Train network to predict pixel colour from a monochrome input



[Courtesy: Zisserman "Self-supervised Learning"]

Colorful Image Colorization, Zhang et al., ECCV 2016

SSL from Images, EX (III): exemplar networks

- Exemplar Networks (Dosovitskiy et al., 2014)
- Perturb/distort image patches, e.g. by cropping and affine transformations
- Train to classify these exemplars as same class

[Courtesy: Zisserman “Self-supervised Learning”]

SSL from Images, EX (IV): masked autoencoder (MAE)

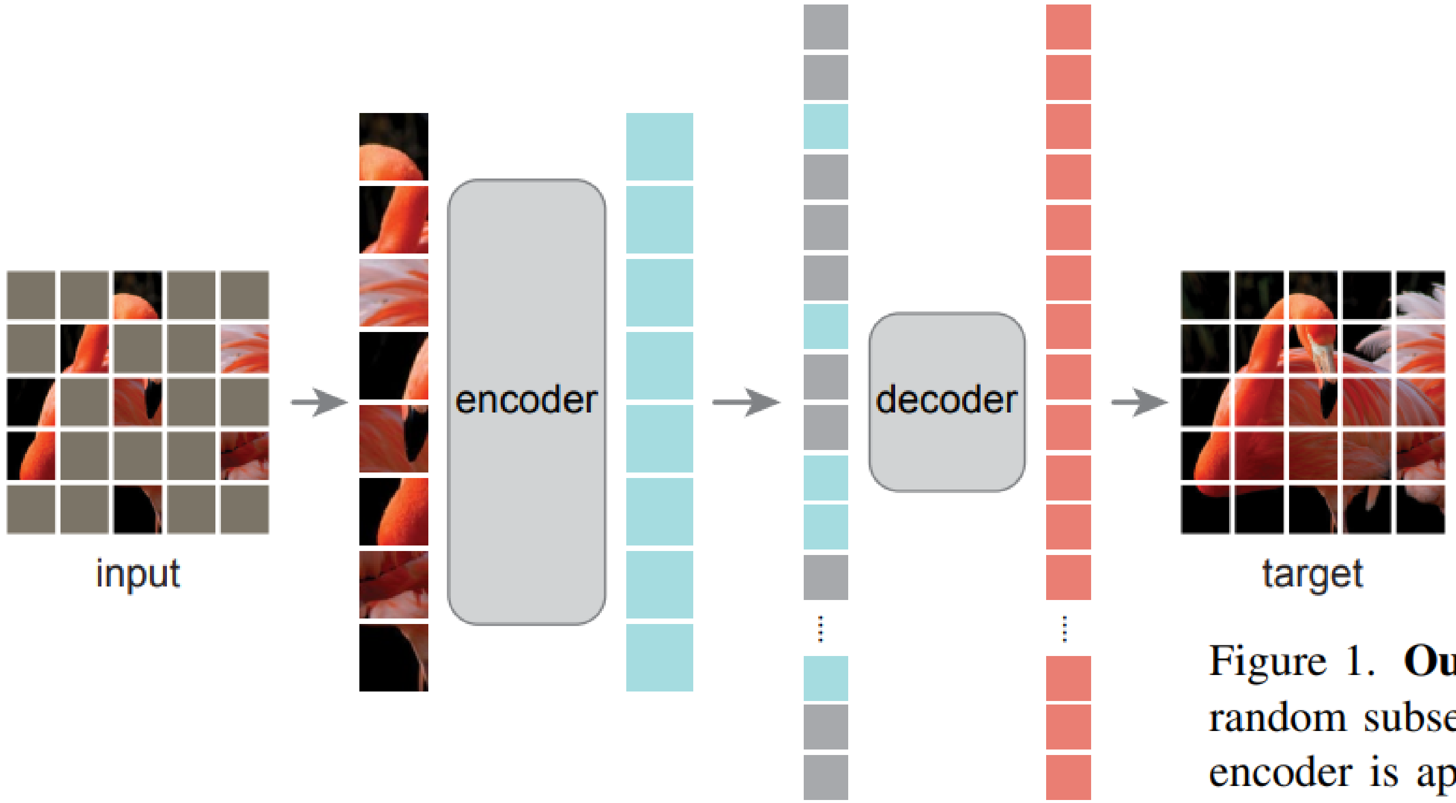
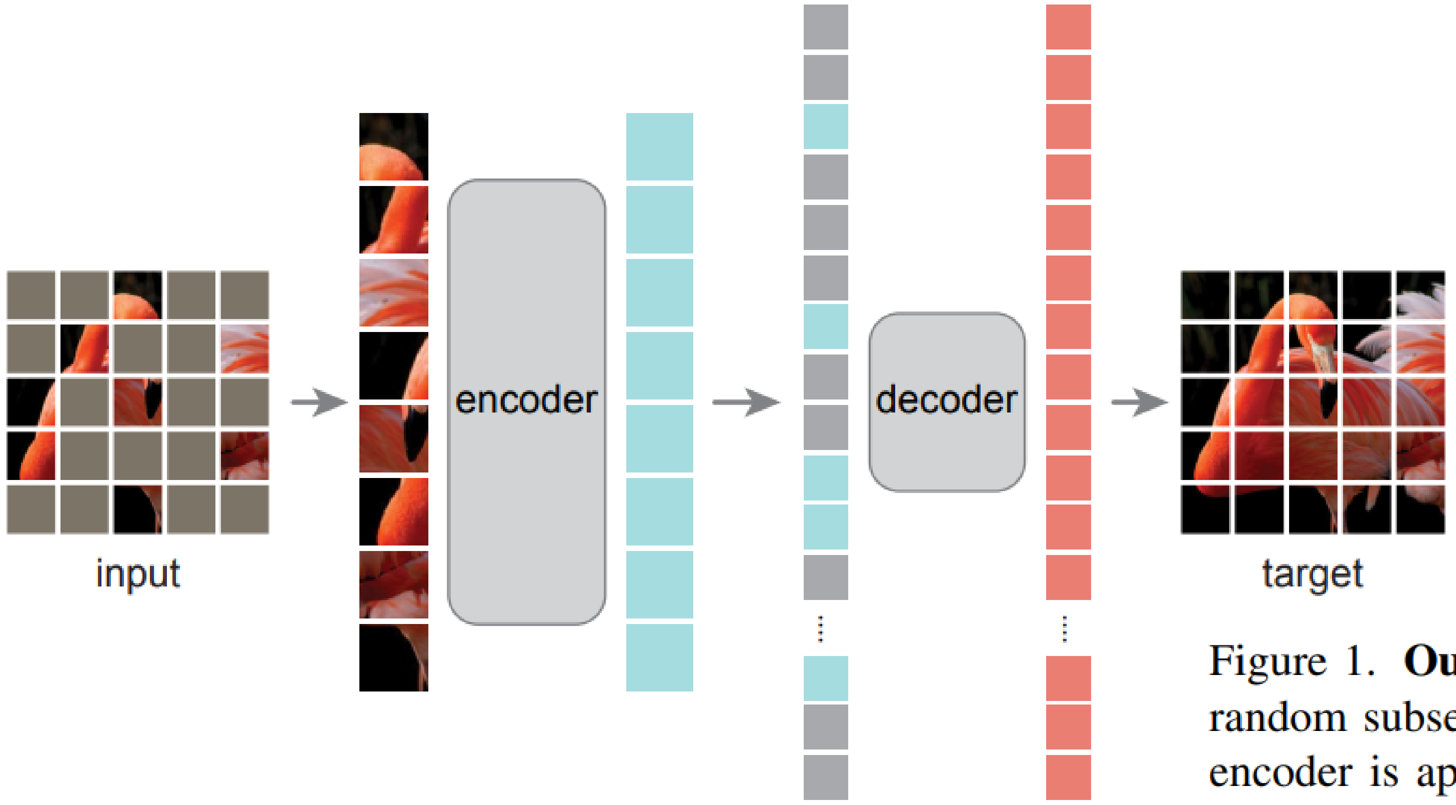


Figure 1. **Our MAE architecture**. During pre-training, a large random subset of image patches (*e.g.*, 75%) is masked out. The encoder is applied to the small subset of *visible patches*. Mask tokens are introduced *after* the encoder, and the full set of encoded patches and mask tokens is processed by a small decoder that reconstructs the original image in pixels. After pre-training, the decoder is discarded and the encoder is applied to uncorrupted images (full sets of patches) for recognition tasks.

[He et al., 2021: Masked Autoencoders Are Scalable Vision Learners]

SSL from Images, EX (IV): masked autoencoder (MAE)



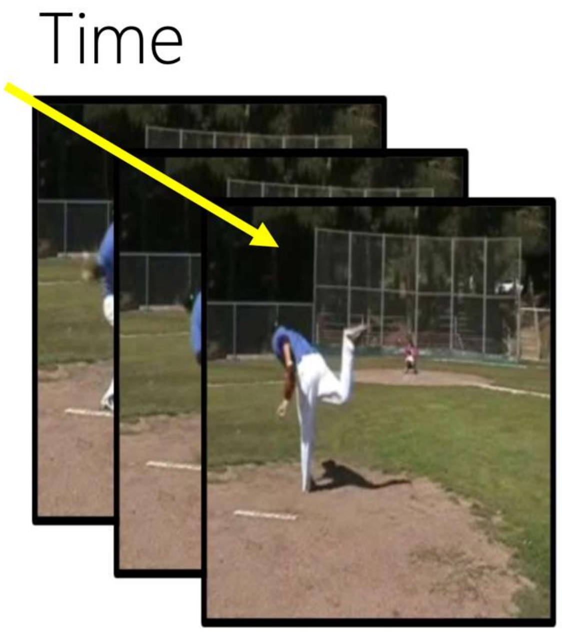
Question: Why is this (75%) much larger than the mask rate in BERT (15%)?

Figure 1. **Our MAE architecture**. During pre-training, a large random subset of image patches (*e.g.*, 75%) is masked out. The encoder is applied to the small subset of *visible patches*. Mask tokens are introduced *after* the encoder, and the full set of encoded patches and mask tokens is processed by a small decoder that reconstructs the original image in pixels. After pre-training, the decoder is discarded and the encoder is applied to uncorrupted images (full sets of patches) for recognition tasks.

[He et al., 2021: Masked Autoencoders Are Scalable Vision Learners]

SSL from Videos

Question: What're your ideas of SSL from videos?



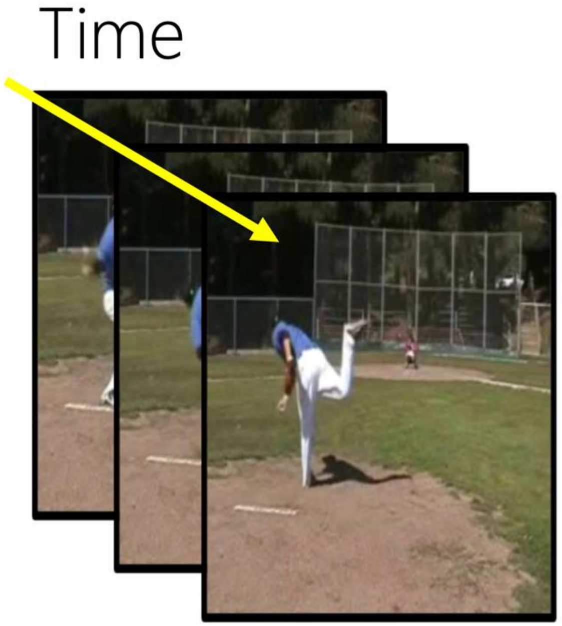
"Sequence" of data

SSL from Videos

Four example tasks:

- Video sequence order
  - Sequential Verification: Is this a valid sequence?





"Sequence" of data

[Courtesy: Zisserman "Self-supervised Learning"]

Wei et al., 2018 Arrow of Time

SSL from Videos

Four example tasks:

- Video sequence order
  - Sequential Verification: Is this a valid sequence?
- Video direction
  - Predict if video playing forwards or backwards

[Courtesy: Zisserman "Self-supervised Learning"]

Wei et al., 2018 Arrow of Time

SSL from Videos

Four example tasks:

- Video sequence order
  - Sequential Verification: Is this a valid sequence?
- Video direction
  - Predict if video playing forwards or backwards
- Video tracking
  - Given a color video, colorize all frames of a gray scale version using a reference frame

[Courtesy: Zisserman “Self-supervised Learning”]

Vondric et al., 2018

SSL from Videos

Four example tasks:

- Video sequence order
  - Sequential Verification: Is this a valid sequence?
- Video direction
  - Predict if video playing forwards or backwards
- Video tracking
  - Given a color video, colorize all frames of a gray scale version using a reference frame
- Video next frame prediction

[Courtesy: Zisserman “Self-supervised Learning”]

Vondric et al., 2018

Key Takeaways

- Self supervision learning
  - Predicting any part of the observations given any available information
  - The prediction task forces models to learn semantic representations
  - Massive/unlimited data supervisions
- SSL for text:
  - Language models: next word prediction
  - BERT text representations: masked language model (MLM)
- SSL for images/videos:
  - Various ways of defining the prediction task

Contrastive Learning

Contrastive learning

- Take a data example x, sample a "positive" sample x_{pos} and "negative" samples x_{neg} in some way
- Then try fit a scoring model such that

$$score(x, x_{pos}) > score(x, x_{neg})$$

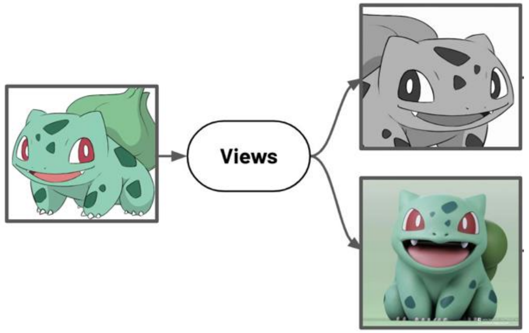
Credit: [CVPR 2021 Tutorial] Leave Those Nets Alone: Advances in Self-Supervised Learning

Contrastive learning

- Take a data example x, sample a "positive" sample x_{pos} and "negative" samples x_{neg} in some way

"positive" sample:

- Data of the same labels
- Data of the same pseudo-labels
- Augmented/distorted version of x
- Data that captures the same target from different views



"negative" sample:

- Randomly sampled data
- Hard negative sample mining

Contrastive learning

- Take a data example x, sample a "positive" sample x_{pos} and "negative" samples x_{neg} in some way
- Then try fit a scoring model such that

$$score(x, x_{pos}) > score(x, x_{neg})$$

Credit: [CVPR 2021 Tutorial] Leave Those Nets Alone: Advances in Self-Supervised Learning

Contrastive learning: Ex 1

Learning a similarity metric discriminatively

Sample a pair of images and compute their distance:

$$D_i = \|x, x_i\|_2$$

If **positive** sample:

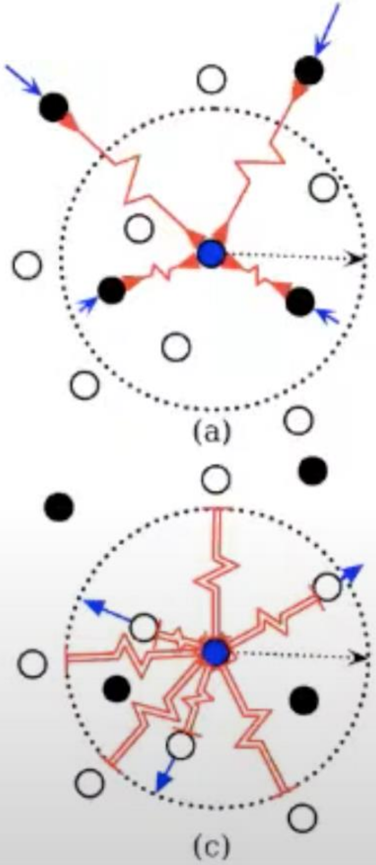
$$L_i = D_i^2$$

x pos

If **negative** sample:

$$L_i = \max\left(0, \epsilon - D_i\right)^2$$

x neg



[Chopra et al., 2005; Hadsell et al., 2006]

Credit: [CVPR 2021 Tutorial] Leave Those Nets Alone: Advances in Self-Supervised Learning

Common contrastive learning functions

- Contrastive loss (Chopra et al. 2005)
- Triplet loss (Schroff et al. 2015; FaceNet)
- Lifted structured loss (Song et al. 2015)
- Multi-class n-pair loss (Sohn 2016)
- Noise contrastive estimation ("NCE"; Gutmann & Hyvarinen 2010)
- InfoNCE (van den Oord, et al. 2018)
- Soft-nearest neighbors loss (Salakhutdinov & Hinton 2007, Frosst et al. 2019)

[Courtesy: Weng & Kim, NeurIPS 2021 tutorial]

Contrastive learning: Ex 2

- SimCSE ("Simple Contrastive learning of Sentence Embeddings"; Gao et al. 2021)
  - Predict a sentence from itself with only dropout noise
  - One sentence gets two different versions of dropout augmentations

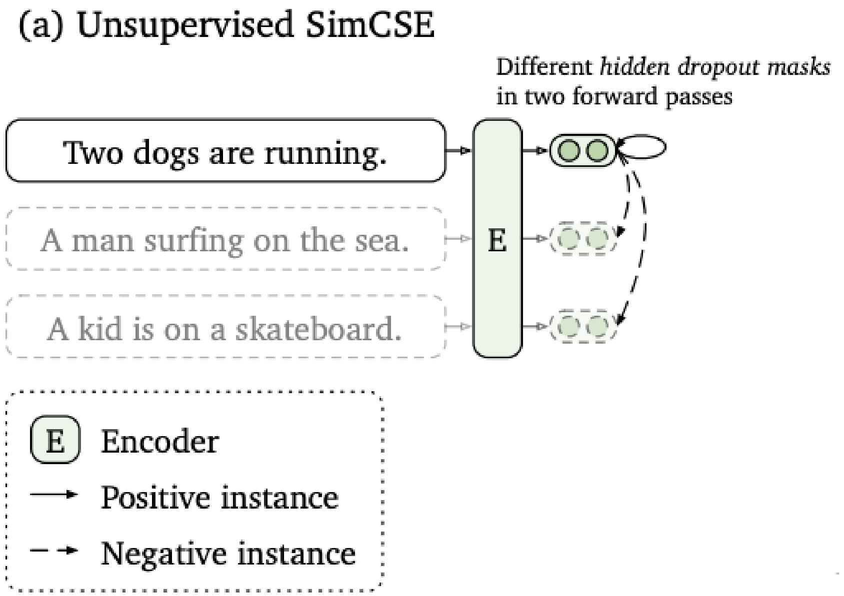


Figure 1: (a) Unsupervised SimCSE predicts the input sentence itself from in-batch negatives, with different hidden dropout masks applied.

Contrastive learning: Ex 2

- SimCSE ("Simple Contrastive learning of Sentence Embeddings"; Gao et al. 2021)
  - Predict a sentence from itself with only dropout noise
  - One sentence gets two different versions of dropout augmentations

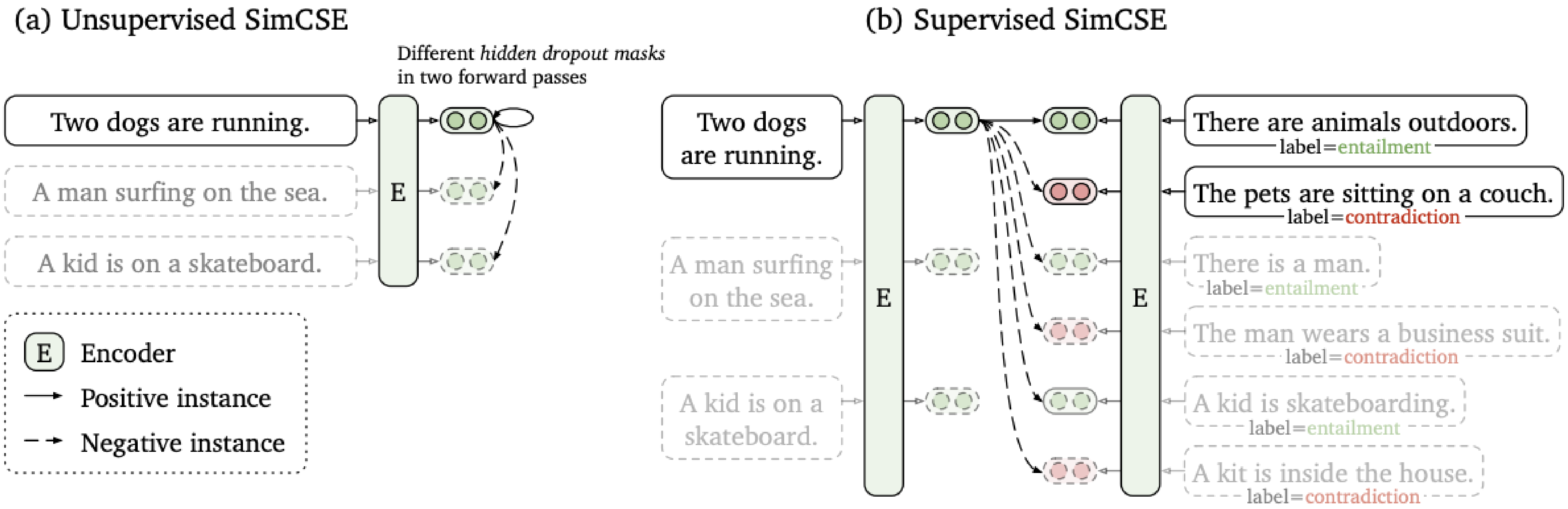
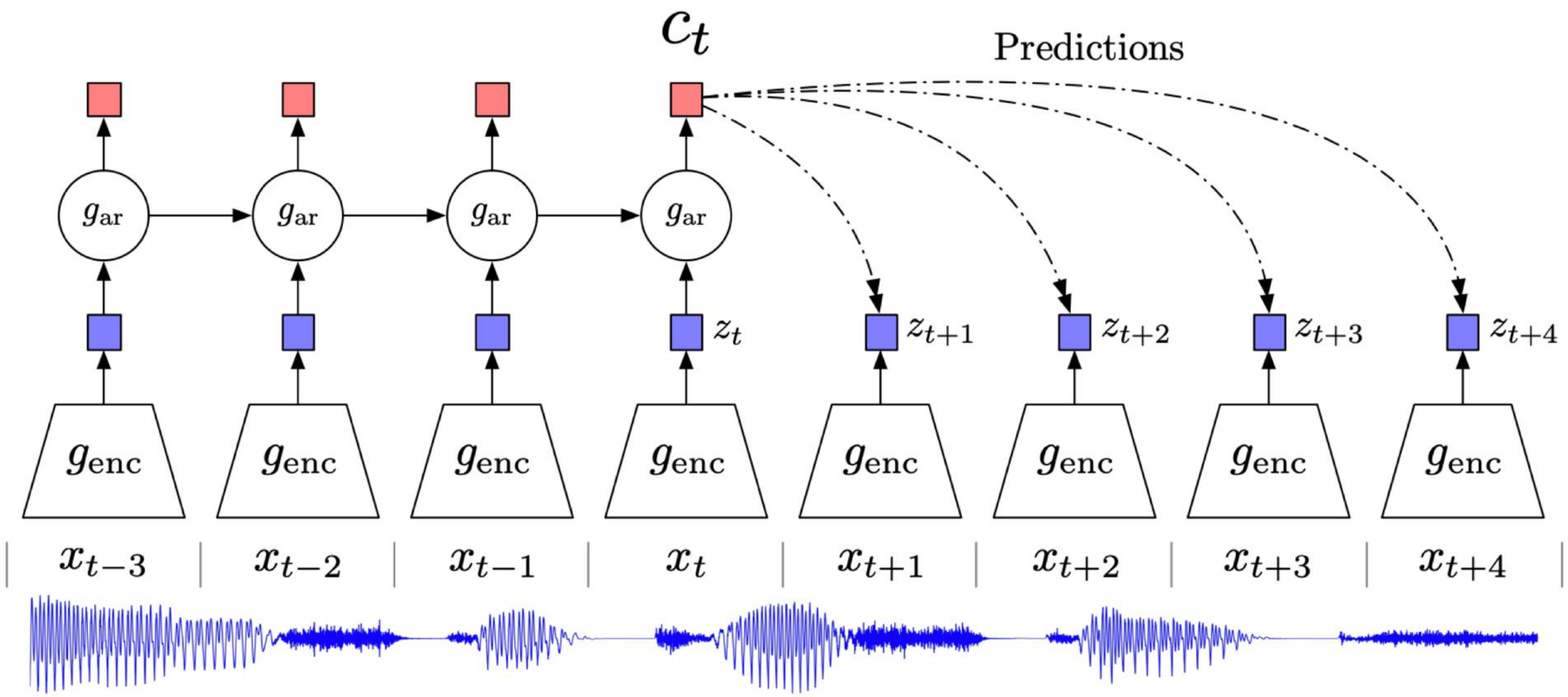


Figure 1: (a) Unsupervised SimCSE predicts the input sentence itself from in-batch negatives, with different hidden dropout masks applied. (b) Supervised SimCSE leverages the NLI datasets and takes the entailment (premise-hypothesis) pairs as positives, and contradiction pairs as well as other in-batch instances as negatives.

Contrastive learning: Ex 3 - InfoNCE

- The CPC model
  - c_t: context representation from history
  - x_{t+k} (or z_{t+k}): future target



[van den Oord et al., “Representation Learning with Contrastive Predictive Coding”]

InfoNCE loss

- Define scoring function $f_k > 0$
- The InfoNCE loss:
  - Given $X = \{$ one positive sample from $p(x_{t+k}|\, c_t)$, $N-1$ negative samples from the negative sampling distribution $p(x_{t+k})$ $\}$

$$\mathcal{L}_{\mathrm{N}} = - \mathop{\mathbb{E}}_{X} \left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)} \right]$$

- InfoNCE is interesting because it's effectively maximizing the **mutual information** between c_t and x_{t+k}

[van den Oord et al., "Representation Learning with Contrastive Predictive Coding"]

Questions?