Machine Learning: Self-Supervised Learning 1

Hao Tang

March 20, 2026

Generalization from one task to another

- Training on one task can sometimes help learn other tasks.
- Two examples
  - Word embeddings (Mikolov *et al.*, 2013)
  - Supervised pre-training in computer vision (Girshick *et al.*, 2014)

- Girshick *et al.* (2014) found that a pre-trained image classifier can be fine-tuned for object detection.
- They named the idea **supervised pre-training**.

	mAP
DPM	33.7
R-CNN pool_5	44.2
R-CNN fc_6	46.2
R-CNN fc_7	44.7
R-CNN fine-tuned pool_5	47.3
R-CNN fine-tuned fc_6	53.1
R-CNN fine-tuned fc_7	54.2

- Girshick *et al.* (2014) found that a pre-trained image classifier can be fine-tuned for object detection.
- They named the idea **supervised pre-training**.

	mAP
DPM	33.7
R-CNN pool_5	44.2
R-CNN fc_6	46.2
R-CNN fc_7	44.7
R-CNN fine-tuned pool_5	47.3
R-CNN fine-tuned fc_6	53.1
R-CNN fine-tuned fc_7	54.2

- Girshick *et al.* (2014) found that a pre-trained image classifier can be fine-tuned for object detection.
- They named the idea **supervised pre-training**.

	mAP
DPM	33.7
R-CNN pool_5	44.2
R-CNN fc_6	46.2
R-CNN fc_7	44.7
R-CNN fine-tuned pool_5	47.3
R-CNN fine-tuned fc_6	53.1
R-CNN fine-tuned fc_7	54.2

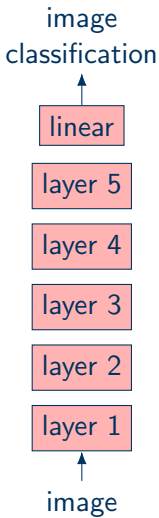


image
classification
linear
layer 5
layer 4
layer 3
layer 2
layer 1
image

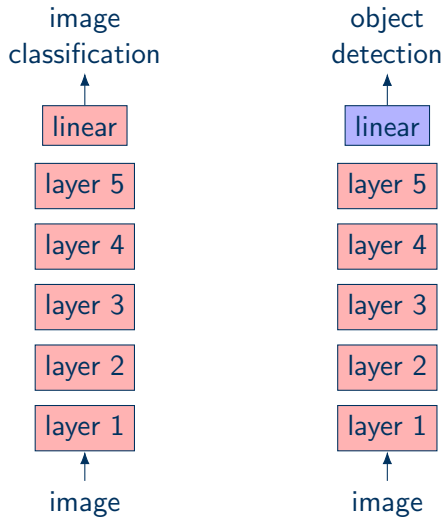


image
classification
linear
layer 5
layer 4
layer 3
layer 2
layer 1
image
object
detection
linear
layer 5
layer 4
layer 3
layer 2
layer 1
image

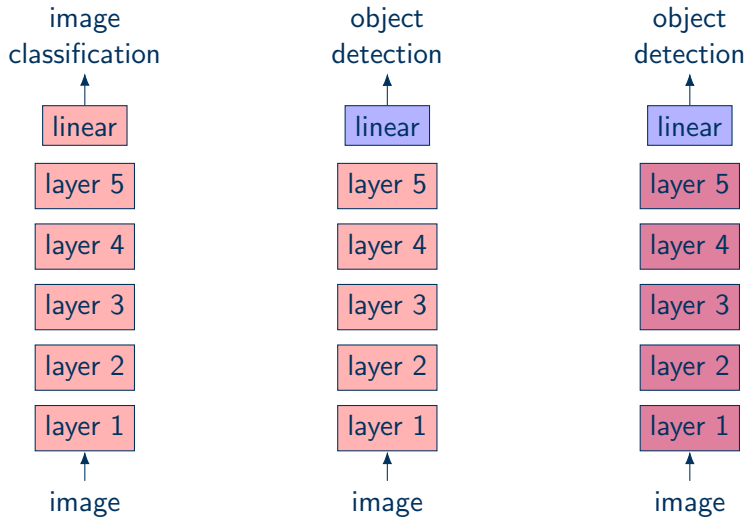
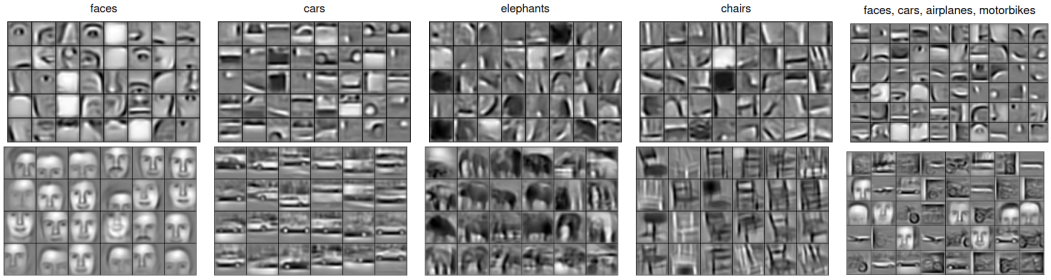


image classification
linear
layer 5
layer 4
layer 3
layer 2
layer 1
image
object detection
linear
layer 5
layer 4
layer 3
layer 2
layer 1
image
object detection
linear
layer 5
layer 4
layer 3
layer 2
layer 1
image

- **Pre-training** is simply regular training but to make clear that there are further training after it.
- **Fine-tuning** is training after pre-training, usually with a loss different from pre-training.
- In other words, fine-tuning is regular training but with an initialization of a trained model.

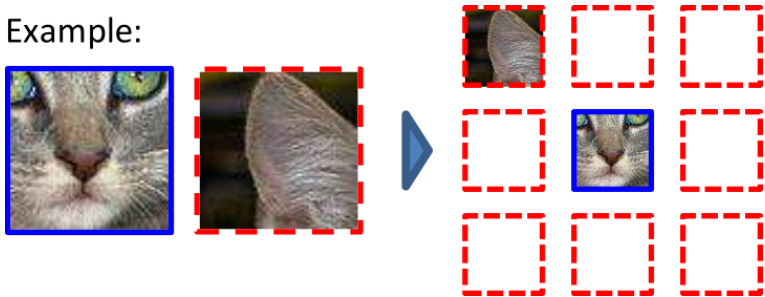
- Training on one task can sometimes help learn other tasks.
- The tasks in pre-training is called the **pretext task**, while the other tasks that might benefit from pre-training are called **downstream tasks**.
- Why does it work?



(Lee *et al.*, 2009)

- Neural networks tend to learn useful intermediate representations to solve a task.
- The intermediate representations can be useful for other tasks too.
- Does pre-training (i.e., the pretext task) need to involve manual labels?

Examples of self-supervised learning



(Doersch *et al.*, 2015)

- The pretext task in this case is to predict the relative position of patches.
- It is more braodly known as context prediction.

	mAP
R-CNN w/o pre-training	39.8
R-CNN self-supervised pre-training	46.3
R-CNN supervised pre-training	54.2

- Inspired by word2vec, Doersch *et al.* (2015) refers to this approach as **self-supervised learning**.
- Why does this work?

- If the models knows _____, then it should be able do well on _____.

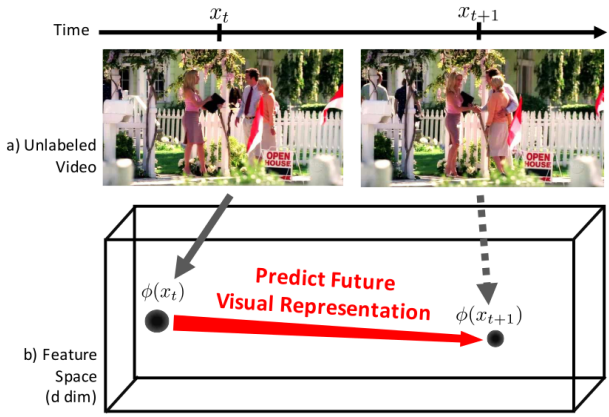
- If the models knows _____, then it should be able do well on _____.
- If the models knows something about images, then it should be able do well on context prediction.

- If the models knows _____, then it should be able do well on _____.
- If the models knows something about images, then it should be able do well on context prediction.
- We train a model to do context prediction and hope that the model can know something about images.

(Pathak *et al.*, 2016)

(Larsson *et al.*, 2016)

Future prediction as self-supervised learning



(Vondrick *et al.*, 2016)

- Predicting the future in the feature space seems like a reasonable pretext task for self-supervised learning.

- To formalize this, we want to train f to predict $\phi(x_{t+1})$ from $\phi(x_t)$, where $\phi(x_t)$ is the feature of x_t.

- The objective is simply

$$\|f(\phi(x_t)) - \phi(x_{t+1})\|^2. \tag{1}$$

- Predicting the future in the feature space seems like a reasonable pretext task for self-supervised learning.

- To formalize this, we want to train f to predict $\phi(x_{t+1})$ from $\phi(x_t)$, where $\phi(x_t)$ is the feature of x_t.

- The objective is simply

$$\|f(\phi(x_t)) - \phi(x_{t+1})\|^2. \tag{1}$$

- However, if we train both f and ϕ to minimize the objective, there are trivial solutions where $\phi(x) = c\mathrm{I}$ for any constant c.

- Predicting the future in the feature space seems like a reasonable pretext task for self-supervised learning.

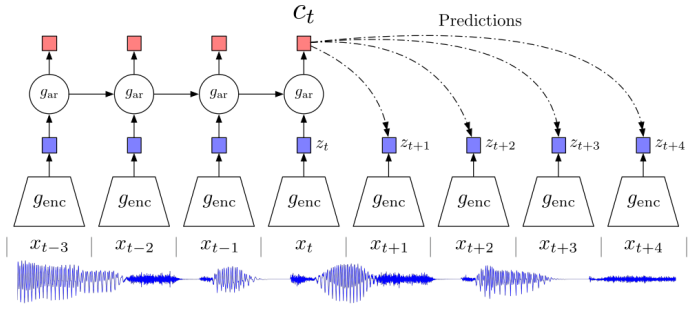
- To formalize this, we want to train f to predict $\phi(x_{t+1})$ from $\phi(x_t)$, where $\phi(x_t)$ is the feature of x_t.

- The objective is simply

$$\|f(\phi(x_t)) - \phi(x_{t+1})\|^2. \tag{1}$$

- However, if we train both f and ϕ to minimize the objective, there are trivial solutions where $\phi(x) = c\mathrm{I}$ for any constant c.

- Vondrick *et al.* (2016) use a pre-trained network for ϕ and only trains f while holding ϕ fixed.



(van den Oord *et al.*, 2018)

- The goal of contrastive predictive coding (CPC) is to predict the future in the feature space.
- It suffers from the same problem, having trivial solutions.
- Instead of predicting the future with mean-squared error, van den Oord *et al.* (2018) adopt a contrastive approach, to distinguish the correct one from others.

- Suppose we want to use c_t to predict z_{t+3}.
- We know that minimizing $\|z_{t+3} - Wc_t\|^2$ leads to a degenerate solution.
- Instead, we want $z_{t+3}^\top Wc_t$ to be high, and $z^\top Wc_t$ to be low for any other z.
- The correct sample, in this case z_{t+3}, is typically called the **positive example**, while the others are called **negative examples**.

- We want $z_{t+3}^\top W c_t$ to be high, and $z^\top W c_t$ to be low for any other z.
- In other words, we want

$$\log \frac{\exp(z_{t+3}^\top W c_t)}{\sum_{z \in N \cup \{z_{t+3}\}} \exp(z^\top W c_t)} \tag{2}$$

  to be high, where N is the set of negative samples.
- The final objective is

$$\sum_{t=1}^{T-3} \log \frac{\exp(z_{t+3}^\top W c_t)}{\sum_{z \in N \cup \{z_{t+3}\}} \exp(z^\top W c_t)}. \tag{3}$$

- The negative examples are typically just all the frames in the batch.

- The number of frames into the future (3 in z_{t+3}) is a *necessary* hyperparameter.

- Language modeling is the task of predicting the next word given the past.
- Given a sentence of k words $t_1, \ldots, t_k$, the objective of language modeling is to maximize

$$p(t_1, \ldots, t_k) = \prod_{i=1}^{k} p(t_i | t_1, \ldots, t_k). \tag{4}$$

- Language modeling is a form of future prediction.

Deep contextualized word representations

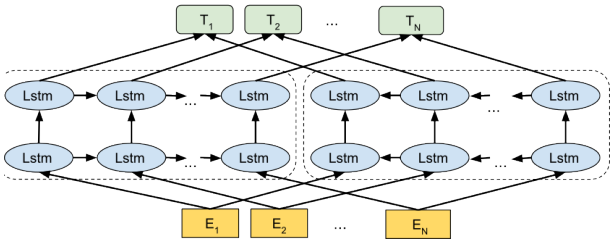
Matthew E. Peters[†], **Mark Neumann**[†], **Mohit Iyyer**[†], **Matt Gardner**[†],
{matthewp,markn,mohiti,mattg}@allenai.org

Christopher Clark[*], **Kenton Lee**[*], **Luke Zettlemoyer**[†*]
{csquared,kentonl,lsz}@cs.washington.edu

[†]Allen Institute for Artificial Intelligence
[*]Paul G. Allen School of Computer Science & Engineering, University of Washington

- ELMo (embeddings from language models) use the same future prediction objective.

- In fact, ELMo actually includes past prediction as well.



(Devlin *et al.*, 2019)

Improving Language Understanding by Generative Pre-Training

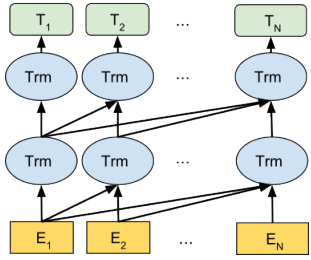
Alec Radford
OpenAI
alec@openai.com

Karthik Narasimhan
OpenAI
karthikn@openai.com

Tim Salimans
OpenAI
tim@openai.com

Ilya Sutskever
OpenAI
ilyasu@openai.com

- GPT uses the same future prediction objective but with Transformers.



(Devlin *et al.*, 2019)

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Jacob Devlin **Ming-Wei Chang** **Kenton Lee** **Kristina Toutanova**
Google AI Language
{jacobdevlin,mingweichang,kentonl,kristout}@google.com

Reference

- Lee et al., Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations, 2009
- Mikolov et al., Distributed representations of words and phrases and their compositionality, 2013
- Girshick et al., Rich feature hierarchies for accurate object detection and semantic segmentation, 2014
- Doersch et al., Unsupervised visual representation learning by context prediction, 2015
- Pathak et al., Context encoders: Feature learning by inpainting, 2016
- Larsson et al., Learning representations for automatic colorization, 2016

- Vondrick et al., Anticipating visual representations from unlabeled video, 2016
- van den Oord et al., Representation learning with contrastive predictive coding, 2018
- Peters et al., Deep contextualized word representations, 2018
- Radford et al., Improving language understanding by generative pre-training, 2018
- Devlin et al., BERT: Pre-training of deep bidirectional Transformers for language understanding, 2019