Unsupervised Deep Learning

Tutorial - Part 2

Alex Graves
gravesa@google.com
DeepMind

Marc'Aurelio Ranzato
ranzato@fb.com
facebook
Artificial Intelligence Research

NeurIPS, 3 December 2018

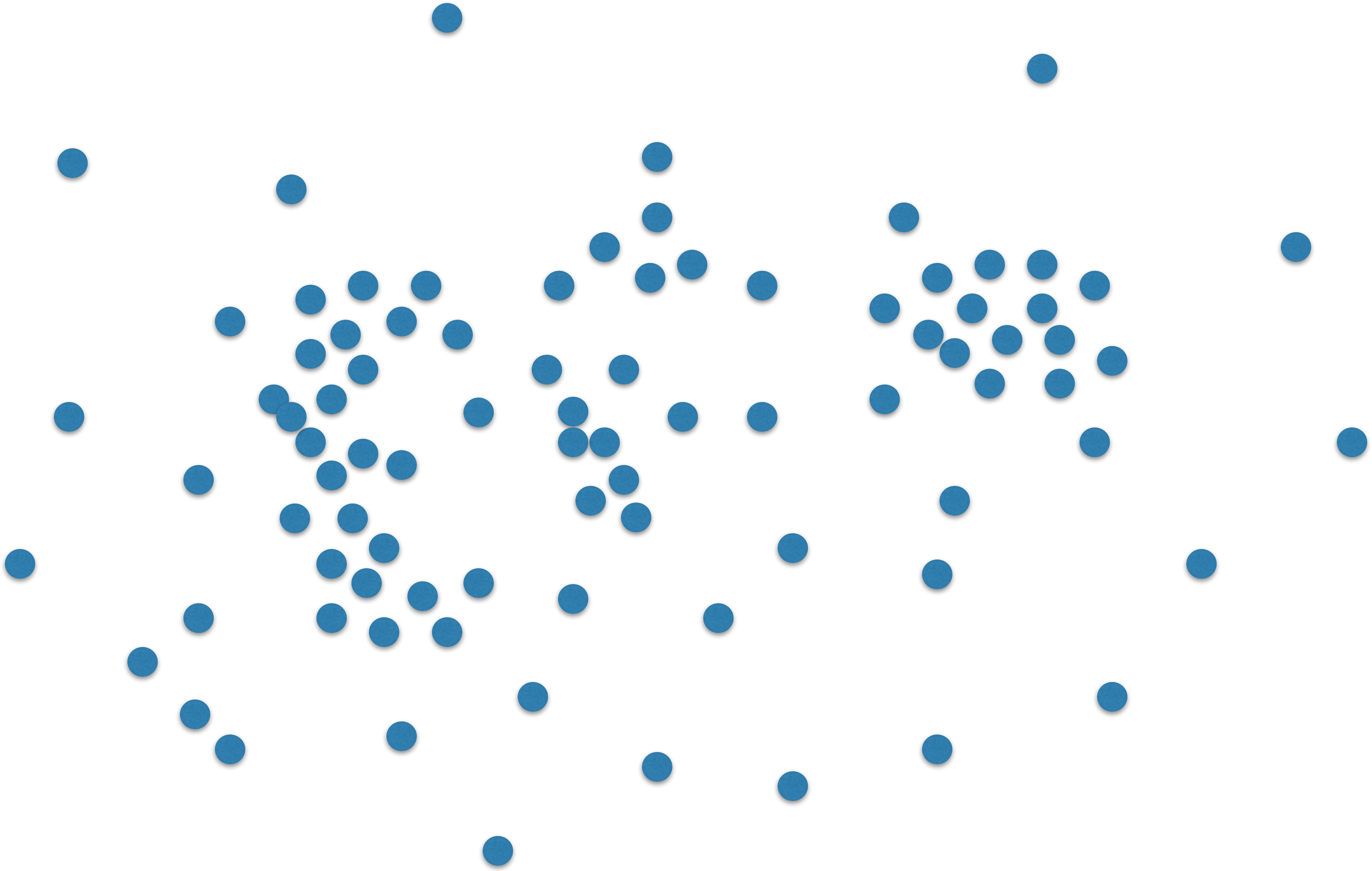
Overview

- Practical Recipes of Unsupervised Learning
  - Learning representations
  - Learning to generate samples
  - Learning to map between two domains
- Open Research Problems

DISCLAIMER

This tutorial is not an exhaustive list of all relevant works!
Goal: overview major research directions in the field and provide pointers for further reading.

Learning Representations: Continuous Case



Toy illustration of the data

Learning Representations

TIP #1: Always "look" at your data before designing your model!

- **mean & covariance analysis**
- **PCA (check eigenvalue decay)**
- **t-sne visualization**

Toy illustration of the data

Learning Representations

$$\begin{bmatrix} 0.1 \\ -2.0 \\ 0.3 \\ 0.7 \\ -0.2 \\ -1.9 \end{bmatrix}$$

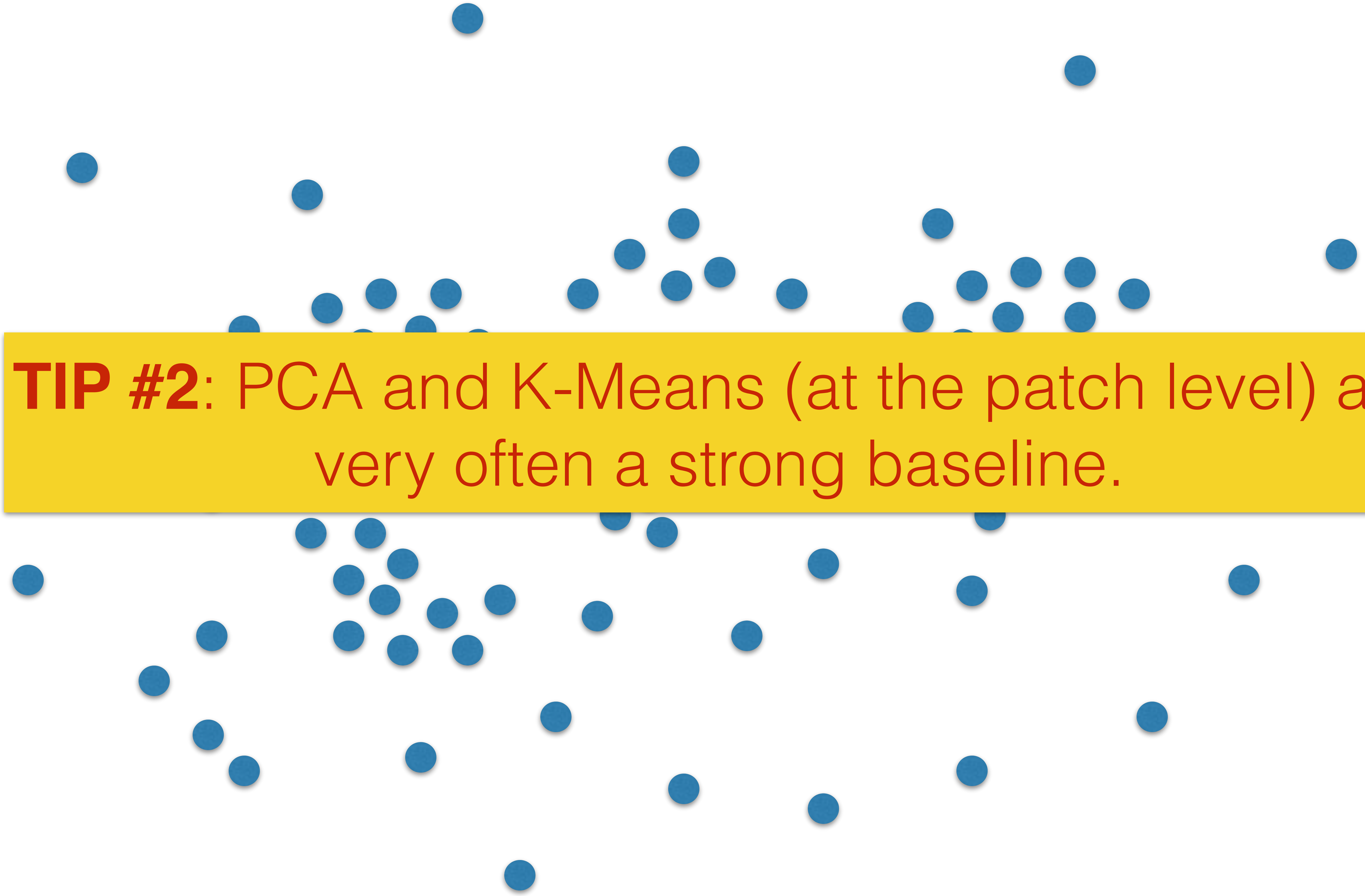
representation learned using unsupervised learning

Task 1: is this person smoking?

Task 2: how likely is this person to have diabetes?

Features are (hopefully) useful in down-stream tasks

Learning Representations

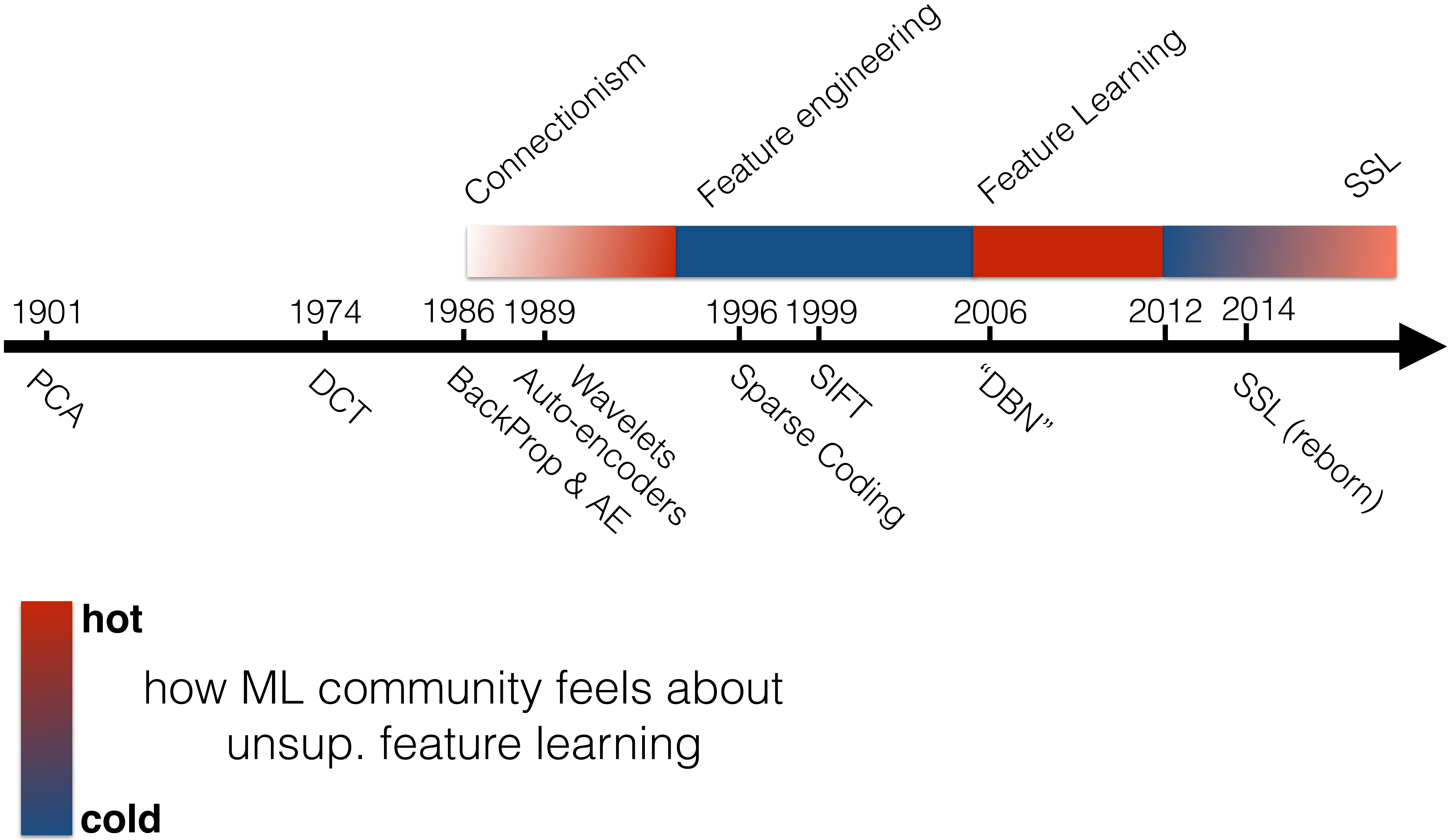


TIP #2: PCA and K-Means (at the patch level) are very often a strong baseline.

Learning **Visual** Representations

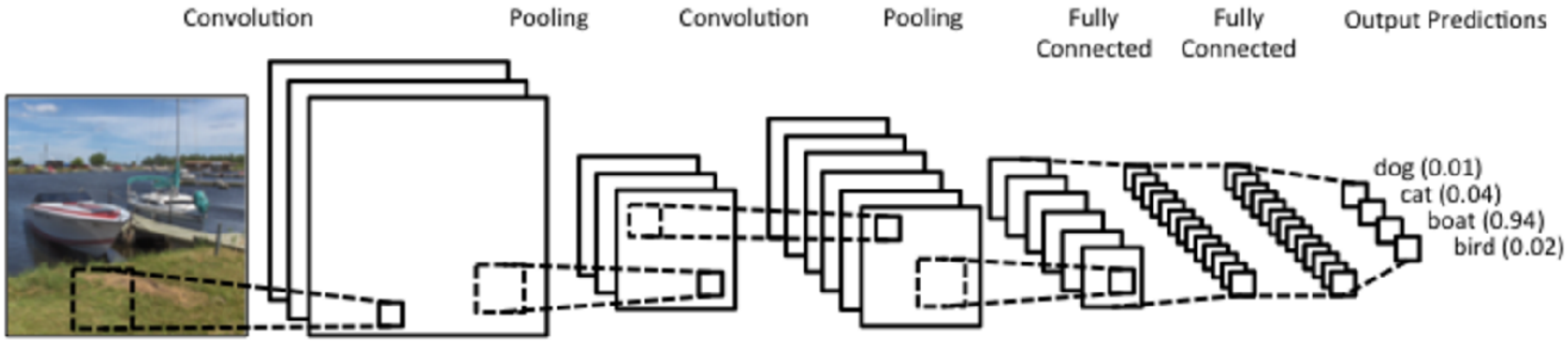
- Brief History
- Self-Supervised Learning
- Other Approaches

Unsup. Feature Learning in Vision



how ML community feels about unsup. feature learning

The Vision Architecture



Convolutional Neural Network

Y. LeCun et al. “Gradient-Based Learning Applied to Document Recognition”, IEEE 1998

A. Krizhevsky et al. “Imagenet classification with CNNs”, NIPS 2012

K. He et al. “Deep Residual Learning for Image Recognition”, CVPR 2016

https://ranzato.github.io/publications/ranzato_deeplearn17_lec1_vision.pdf

Credit for figure:: https://towardsdatascience.com/build-your-own-convolution-neural-network-in-5-mins-4217c2cf964f

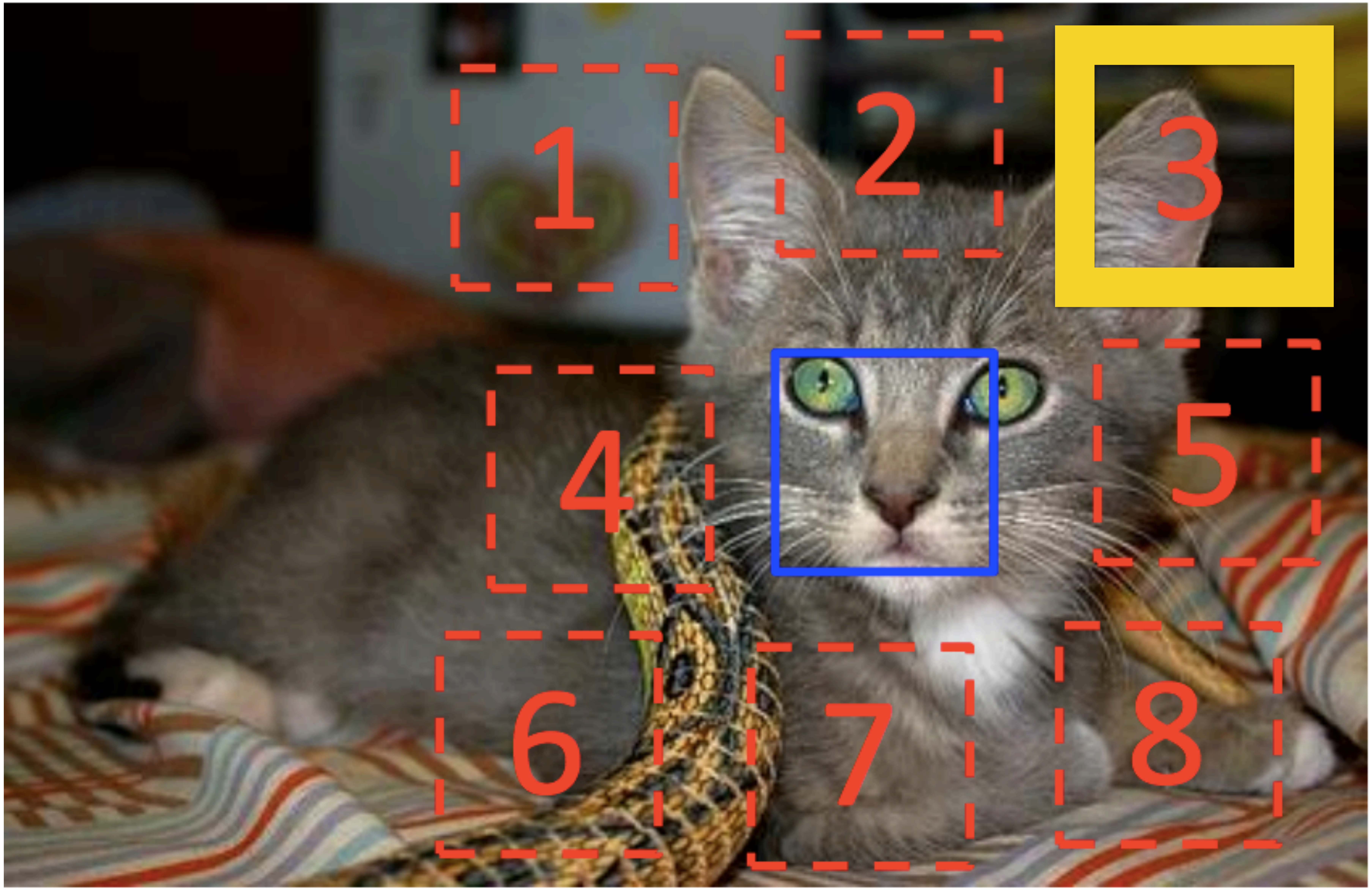
Self-Supervised Learning

- Unsupervised learning is hard: model has to reconstruct high-dimensional input.
- With domain expertise define a prediction task which requires some semantic understanding.
  - conditional prediction (less uncertainty, less high-dimensional)
  - often times, original regression is turned into a classification task

SSL on Static Images: Example

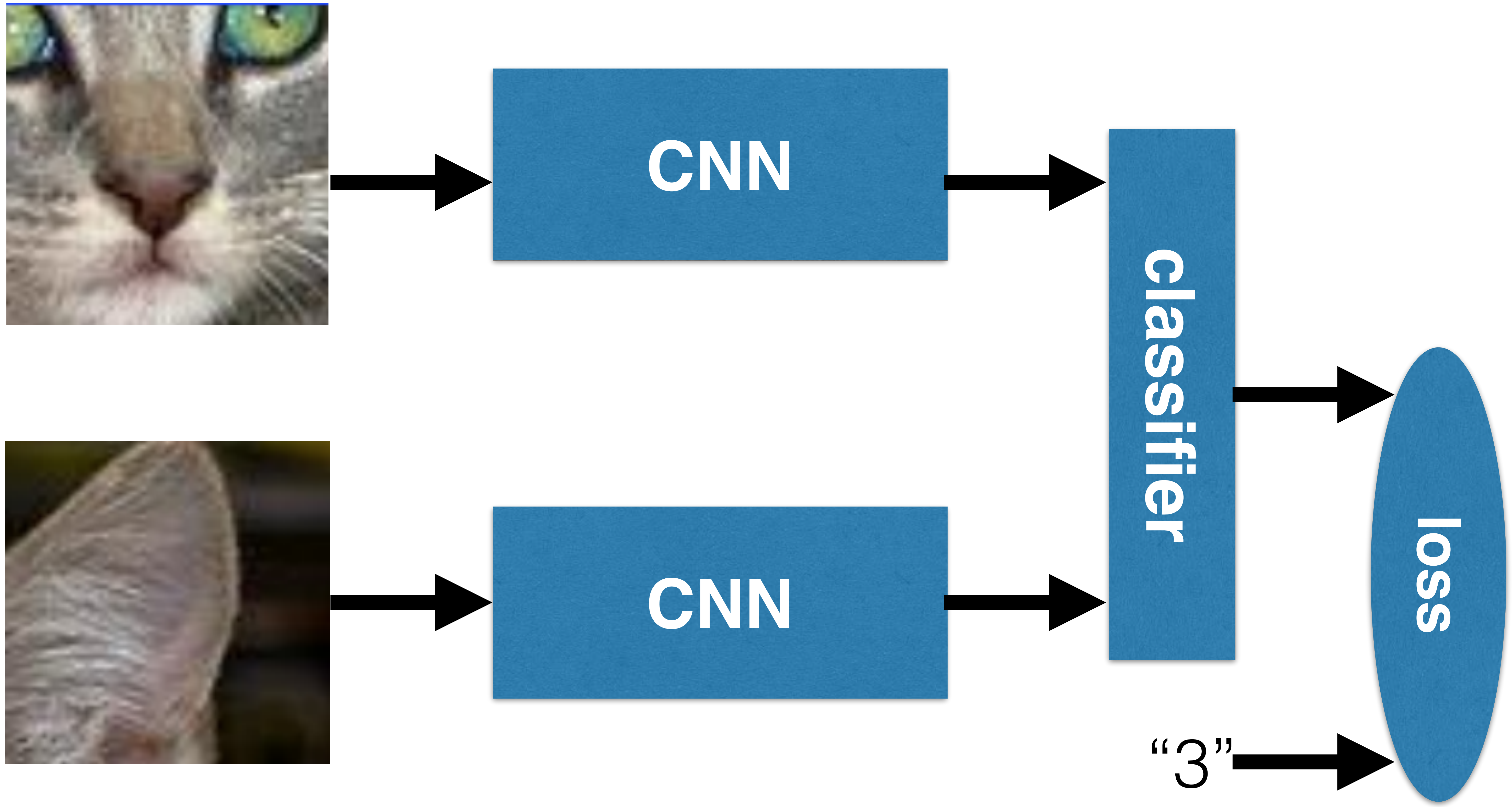

Input: two image patches from the same image.
Task: predict their spatial relationship.

C. Doersch et al. "Unsupervised Visual Representation Learning by Context Prediction", ICCV 2015



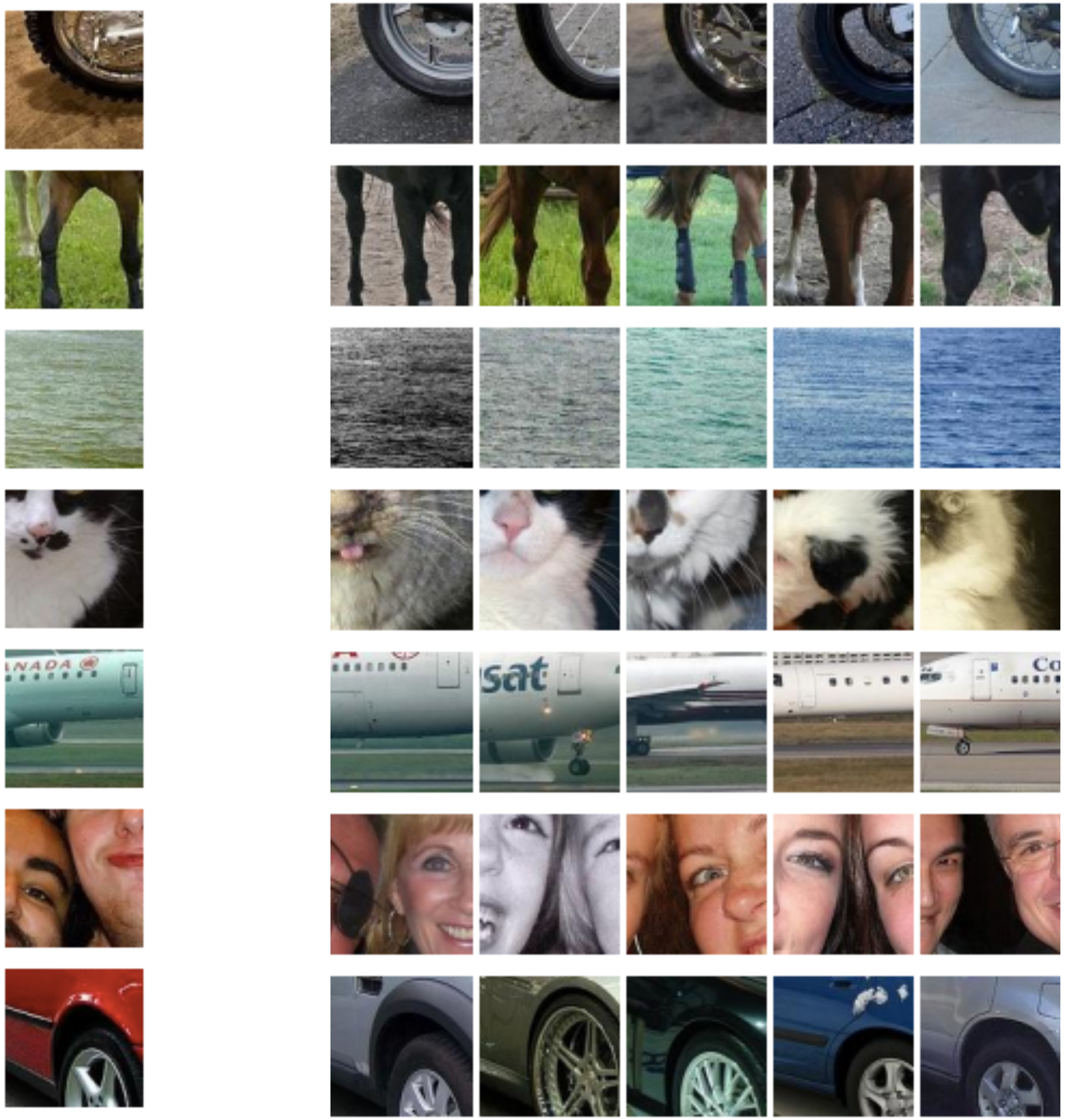
C. Doersch et al. “Unsupervised Visual Representation Learning by Context Prediction”, ICCV 2015

SSL on Static Images: Example



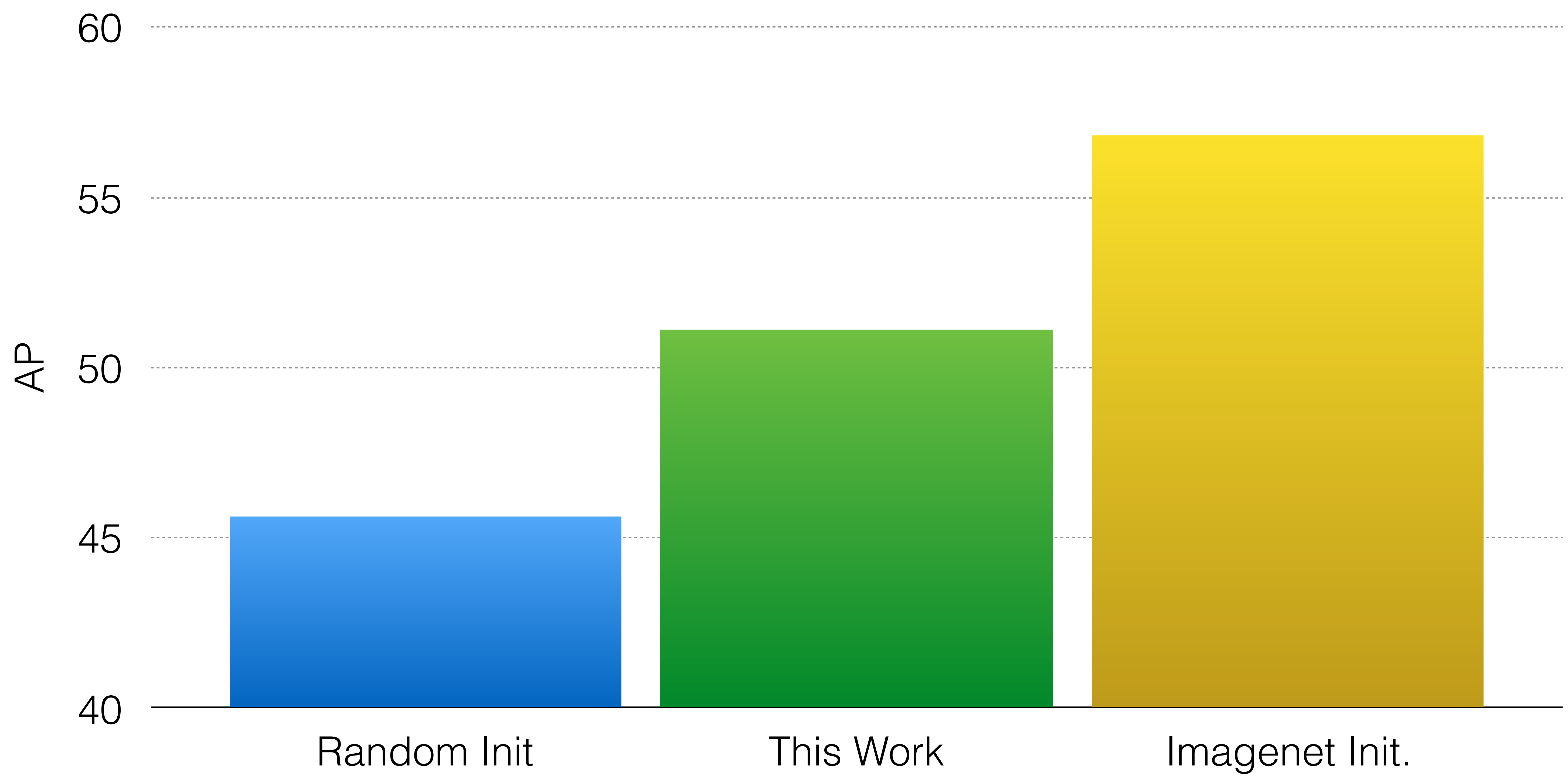
C. Doersch et al. "Unsupervised Visual Representation Learning by Context Prediction", ICCV 2015

Input Nearest Neighbors in Feature Space



C. Doersch et al. “Unsupervised Visual Representation Learning by Context Prediction”, ICCV 2015

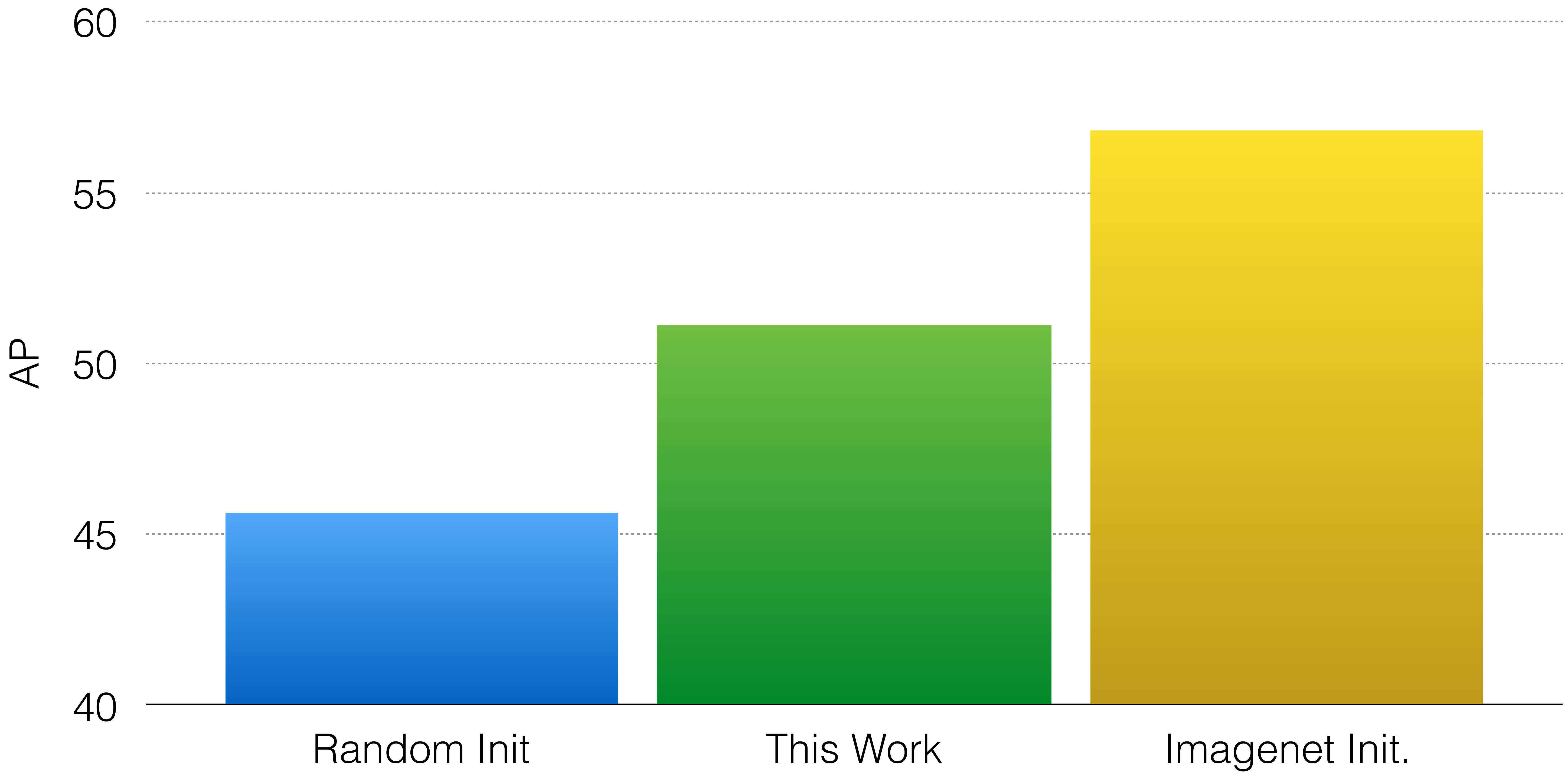
Pascal VOC Detection



C. Doersch et al. “Unsupervised Visual Representation Learning by Context Prediction”, ICCV 2015

Pascal VOC Detection

K. He et al. "Rethinking ImageNet pretraining", arXiv 2018 shows that with better normalization and with longer training, random initialization works as well as ImageNet pretraining!



C. Doersch et al. "Unsupervised Visual Representation Learning by Context Prediction", ICCV 2015

SSL on Static Images: Other Examples

- Predict color from gray scale values.
  R. Zhang et al. "Colorful Image Colorization", ECCV 2016
- Predict image rotation
  S. Gidaris et al. "Unsupervised Representation Learning by Predicting Image Rotations", ICLR 2018

TIP #3: Often times, you can learn features without explicitly predicting pixel values.

TIP #4: If you are OK using domain knowledge, you can learn using a variety of auxiliary tasks.

SSL on **Videos**: Example

- Predict whether the video snippet is playing **forward** or **backward**.
- Requires to understand gravity, causality, friction, …

FWD

D. Wei et al. "Self-supervision using the arrow of time", CVPR 2018

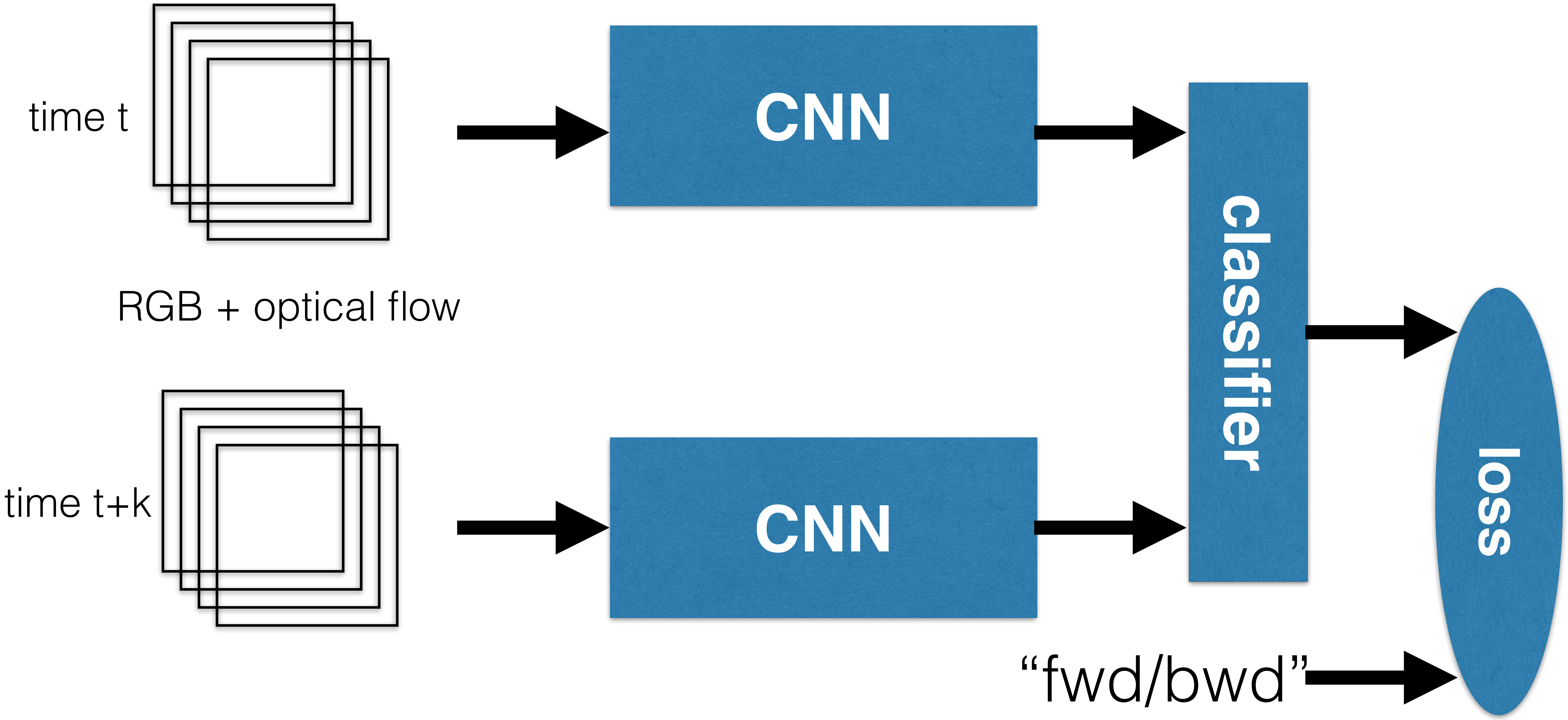
SSL on **Videos**: Example

- Predict whether the video snippet is playing **forward** or **backward**.
- Requires to understand gravity, causality, friction, …

BWD

D. Wei et al. "Self-supervision using the arrow of time", CVPR 2018

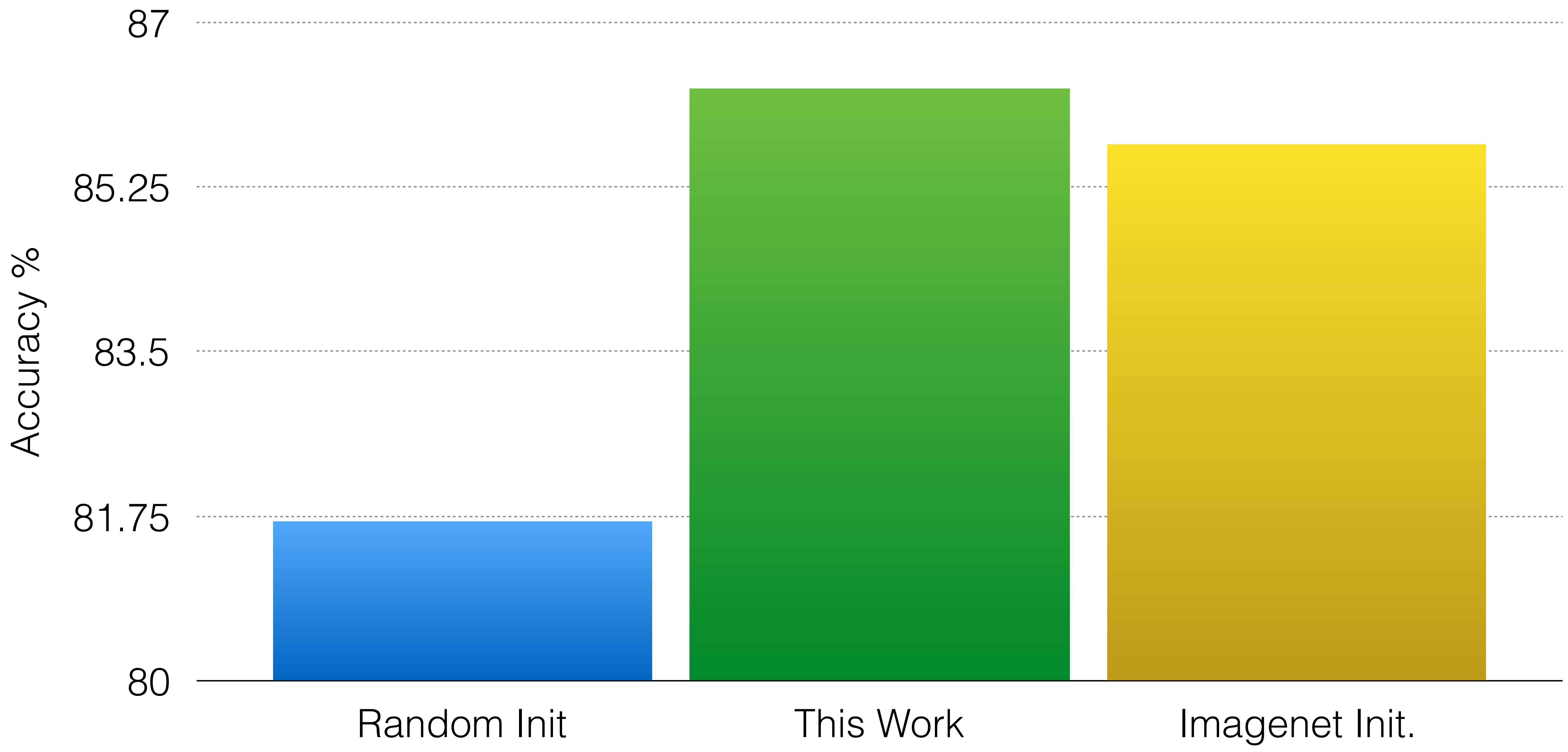
SSL on **Videos**: Example



D. Wei et al. “Self-supervision using the arrow of time”, CVPR 2018

UCF101 Action Recognition

First train using SSL, and then finetune on the task.



D. Wei et al. “Self-supervision using the arrow of time”, CVPR 2018

SSL: Other Examples

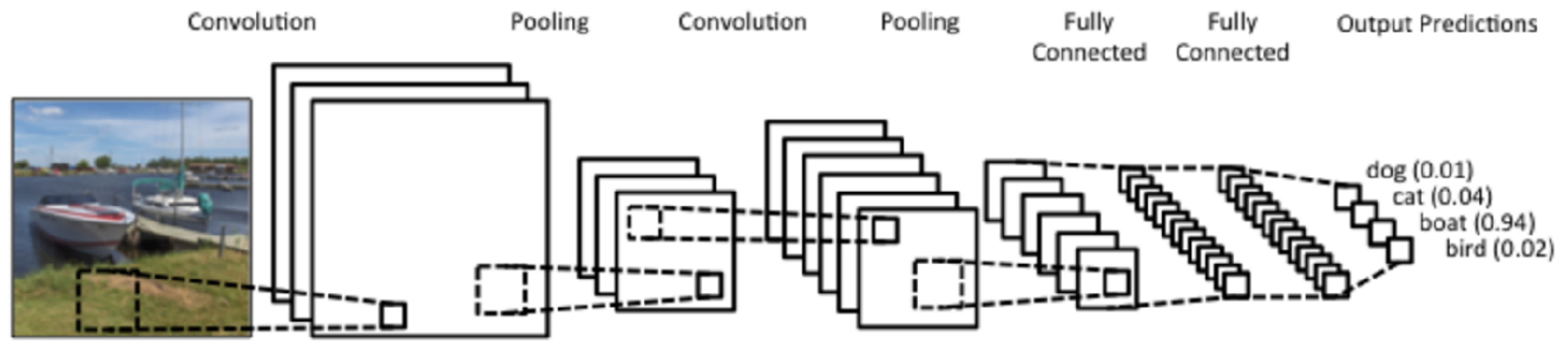
- Learn features by colorizing video sequences.
  C. Vondrik et al. “Tracking emerges from colorizing videos”, ECCV 2018
- Predict whether and how frames are shuffled
  I. Misra et al. “Shuffle and laern: unsupervised learning using temporal order verification”, ECCV 2016
- Future frame prediction
  E. Denton et al. “Unsupervised learning of disentangled representations from video”, NIPS 2017
- Predict one modality from the other
  V. de Sa “Learning classification from unlabeled data”, NIPS 1994
  …
  R. Arandjelovic et al. “Object that sound”, ECCV 2018

Learning **Visual** Representations

- Brief History
- Self-Supervised Learning
- **Other Approaches**

Learning by Clustering

- CNN architecture has many good inductive biases, such as:
  - spatio-temporal stationarity,
  - scale invariance,
  - compositionality, etc.



- (Small) random filters have orientation-frequency selectivity.
- As a result, even randomly initialized CNNs extract non-trivial features.

Learning by Clustering

Randomly initialize the CNN.

Repeat:

1. Extract features from each image and run K-Means in feature space.

2. Train the CNN in supervised mode to predict the cluster id associated to each image (1 epoch).

M. Caron et al. “Deep clustering for unsupervised learning of visual features”, ECCV 2018

Learning by Clustering

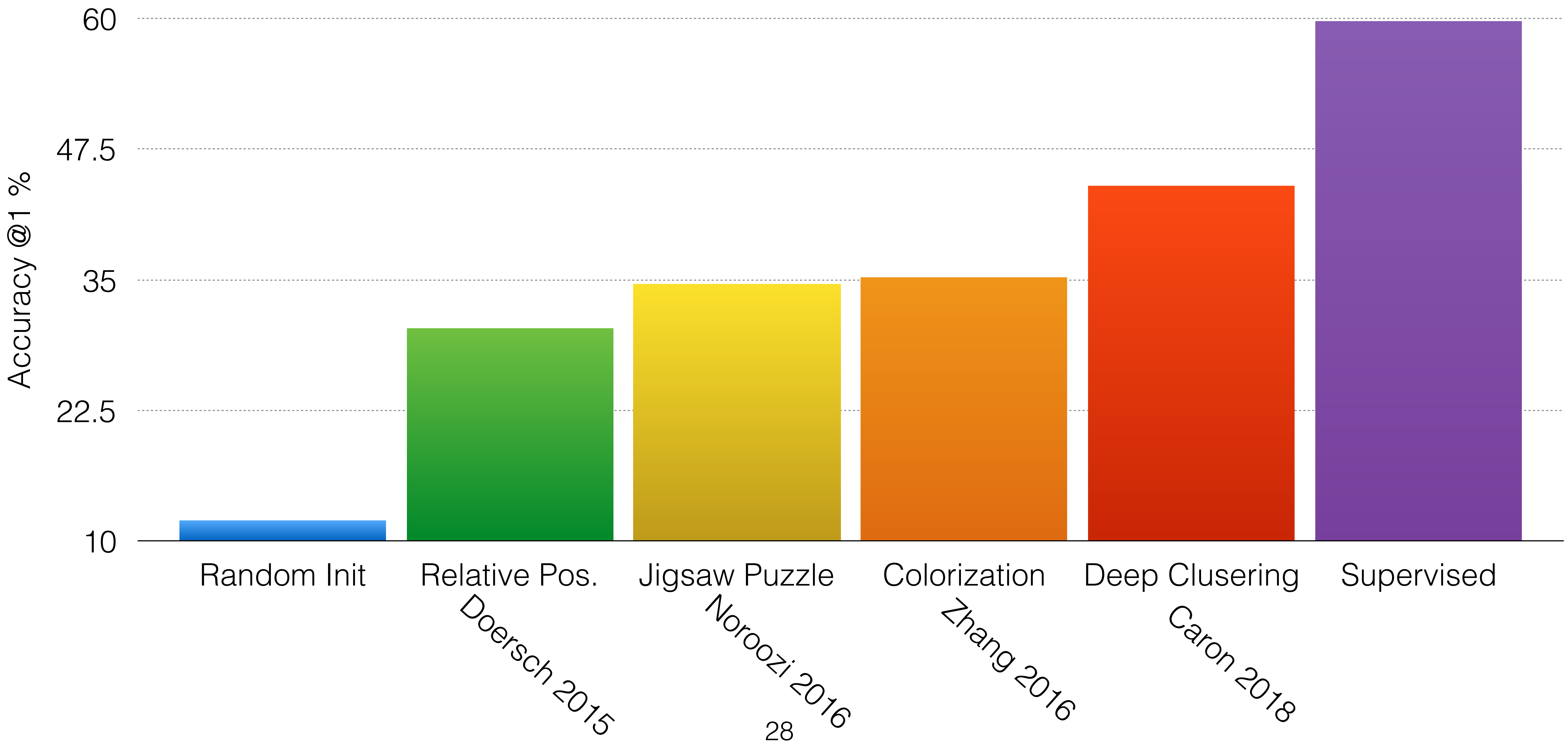
Caveat: watch out for cheating…

- cluster collapsing (re-assign images to empty clusters)
- equalize clusters at training time

M. Caron et al. “Deep clustering for unsupervised learning of visual features”, ECCV 2018

ImageNet Classification

First train unsupervised, then train MLP with supervision using unsupervised features.



Conclusions on Unsupervised Learning of Visual Features

- In general, still a seizable gap between unsupervised feature learning and supervised learning in vision.
- Pixel prediction is hard, many recent approaches define auxiliary classification tasks.
- Domain knowledge can inform the design of tasks that require some level of semantic understanding.
- Network will “cheat” if you are not careful:
    - check for trivial solutions
    - check for biases and artifacts in the data

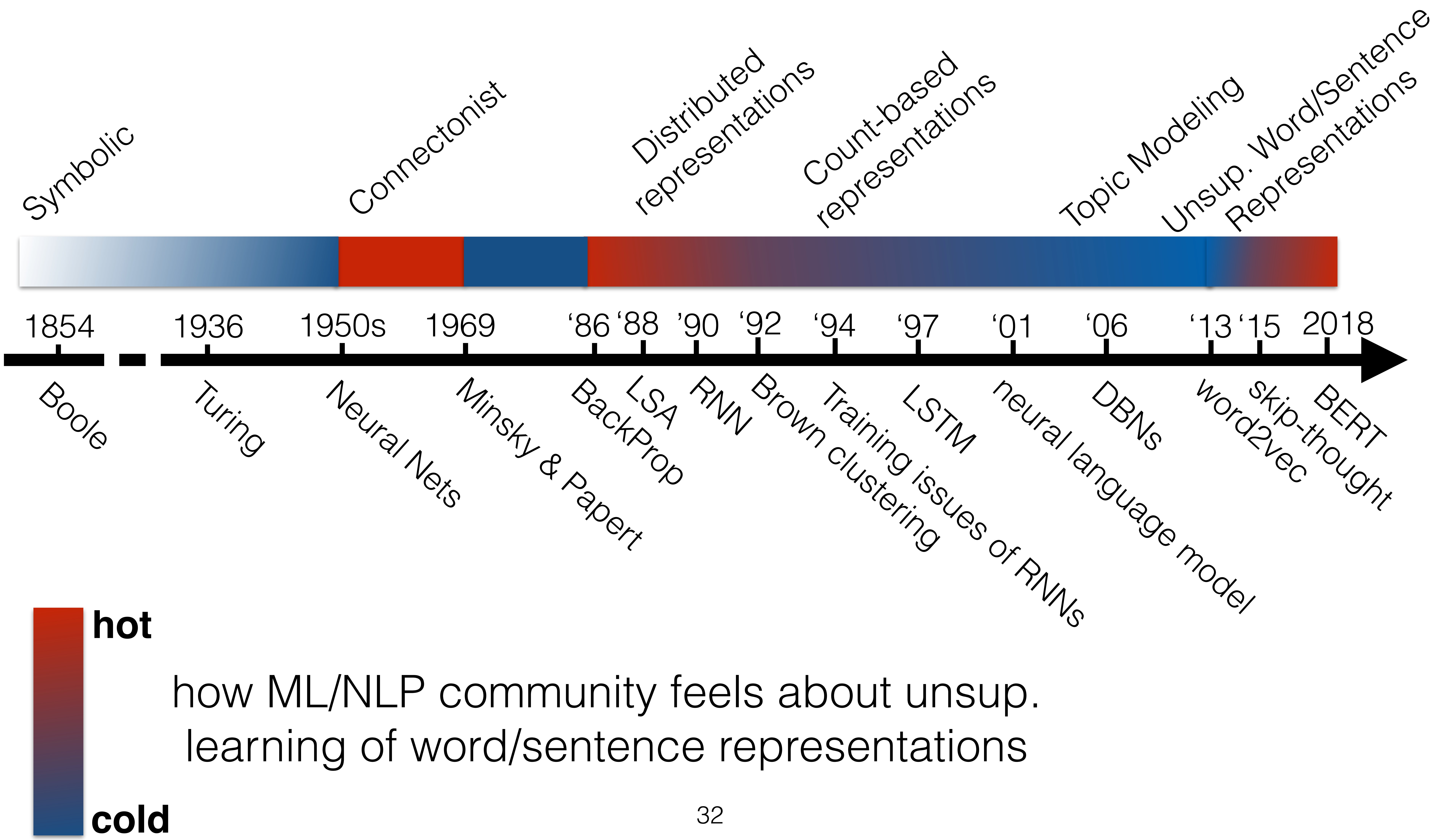
Overview

- Practical Recipes of Unsupervised Learning
    - **Learning representations:** continuous / **discrete**
    - Learning to generate samples: continuous / discrete
    - Learning to map between two domains: continuous / discrete
- Open Research Problems

Vision <—> NLP

- Atomic unit:
    - a word in NLP carries a lot of information.
    - a pixel value in Vision carries negligible information
- Nature of the signal:
    - discrete in NLP: search is hard but modeling of uncertainty is easy.
    - continuous in Vision: search is easy but modeling of uncertainty is hard.

Unsup. Feature Learning in NLP



how ML/NLP community feels about unsup. learning of word/sentence representations

word2vec

"All of the sudden a **cat** jumped from a tree to chase a mouse."

The meaning of a word is determined by its context.

T. Mikolov et al. "Efficient estimation of word representations in vector space" arXiv 2013

word2vec

"All of the sudden a ____ jumped from a tree to chase a mouse."

The meaning of a word is determined by its context.

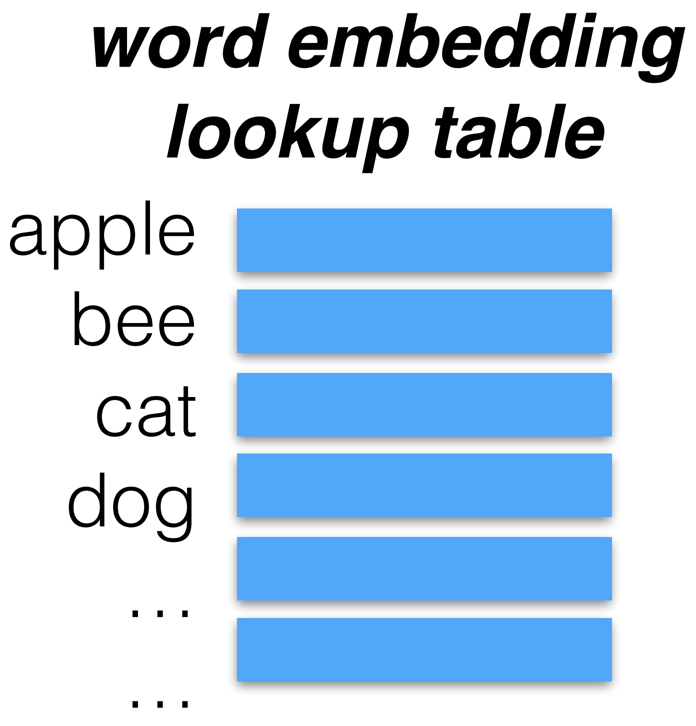
T. Mikolov et al. "Efficient estimation of word representations in vector space" arXiv 2013

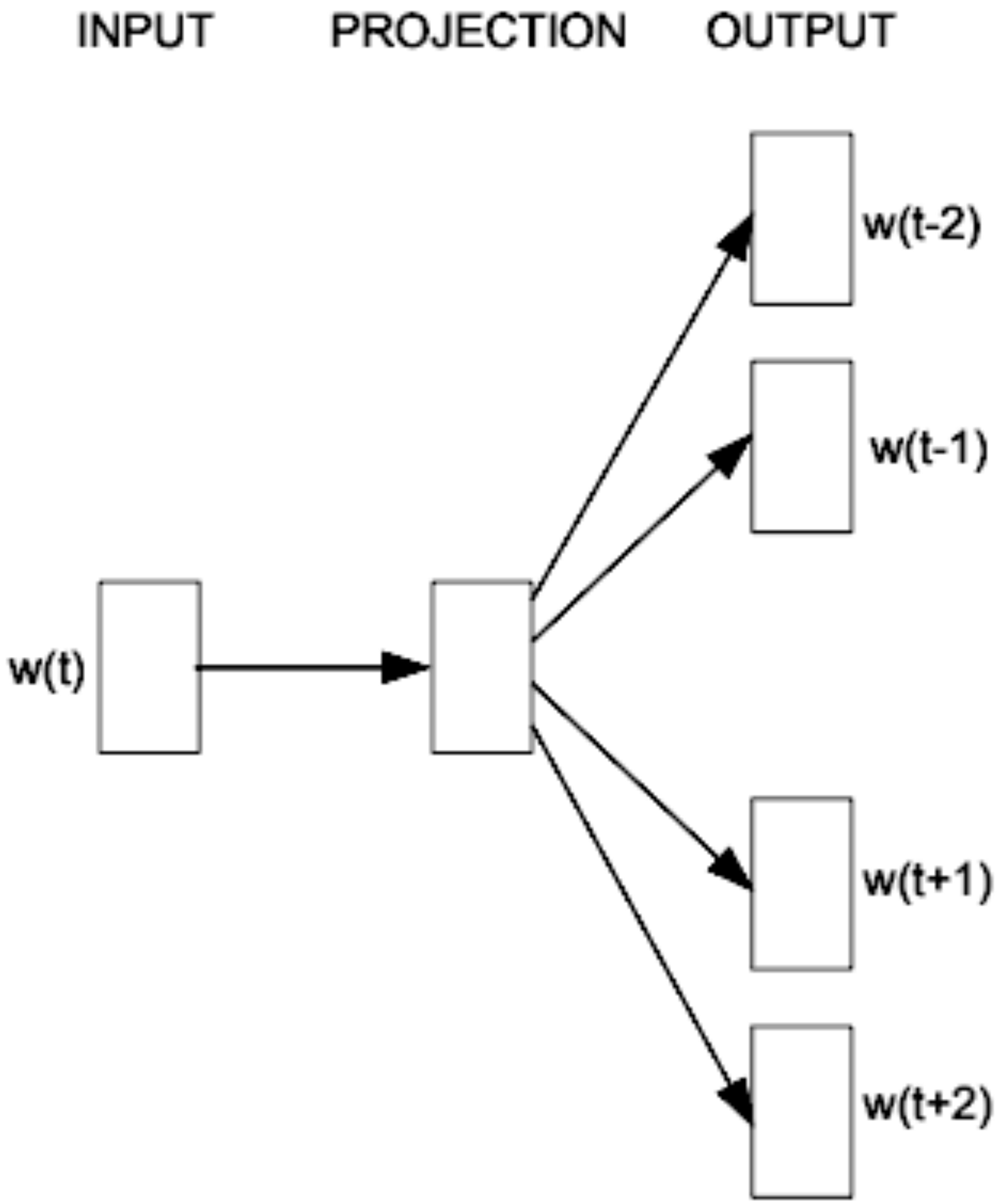
word2vec

“All of the sudden a **kitty** jumped from a tree to chase a mouse.”

The meaning of a word is determined by its context.
Two words mean similar things if they have similar context.

T. Mikolov et al. “Efficient estimation of word representations in vector space” arXiv 2013

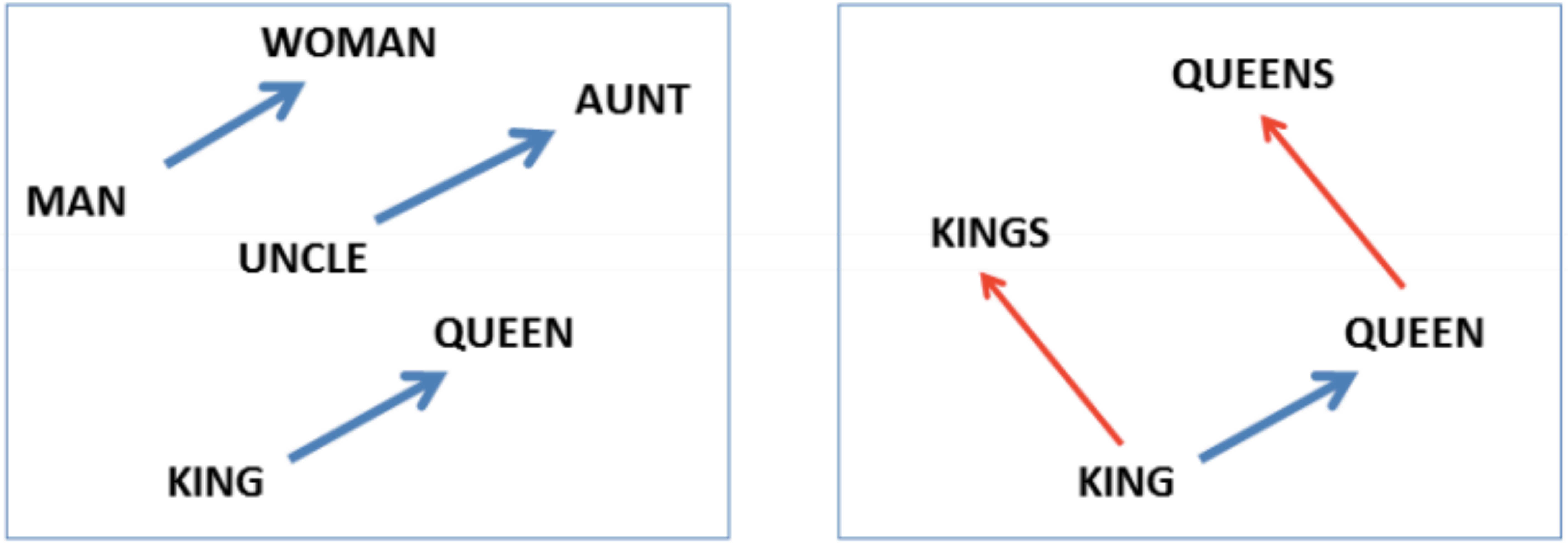




The meaning of a word is determined by its context.
Two words mean similar things if they have similar context.

T. Mikolov et al. “Efficient estimation of word representations in vector space” arXiv 2013

Linguistic Regularities in Word Vector Space



- The word vector space implicitly encodes many regularities among words

credit T. Mikolov from https://drive.google.com/file/d/0B7XkCwpI5KDYRWRnd1RzWXQ2TWc/edit

Recap word2vec

- Word embeddings are useful to:
  - understand similarity between words
  - convert *any discrete input* into continuous -> ML
- Learning leverages large amounts of unlabeled data.
- It’s a very simple factorization model (shallow).
- There are very efficient tools publicly available.

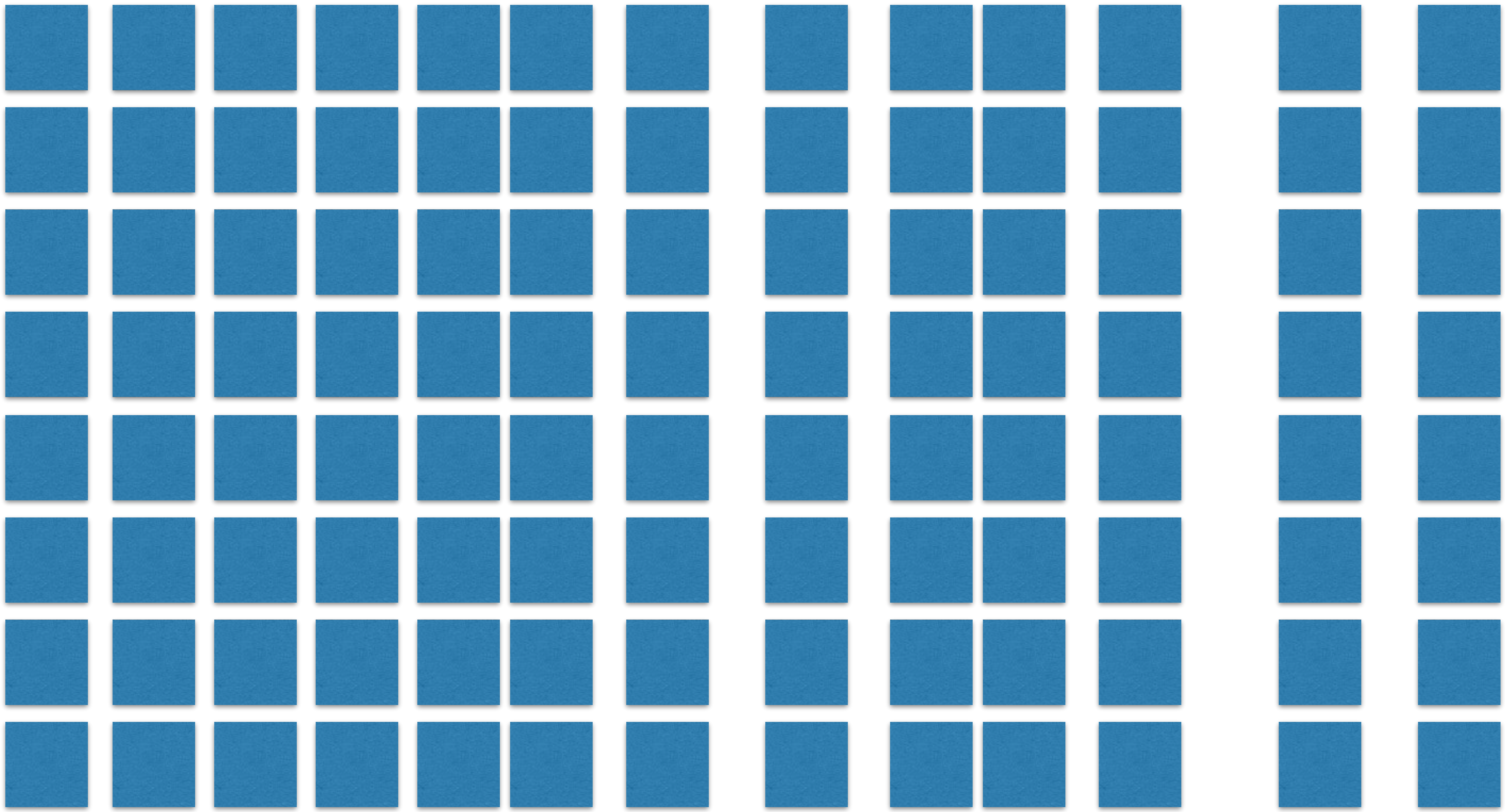
https://fasttext.cc/

Joulin et al. “Bag of tricks for efficient text classification” ACL 2016

Representing Sentences

- word2vec can be extended to small phrases, but not much beyond that.
- Sentence representation needs to leverage compositionality.
- A lot of work on learning unsupervised sentence representations (auto-encoding / prediction of nearby sentences).

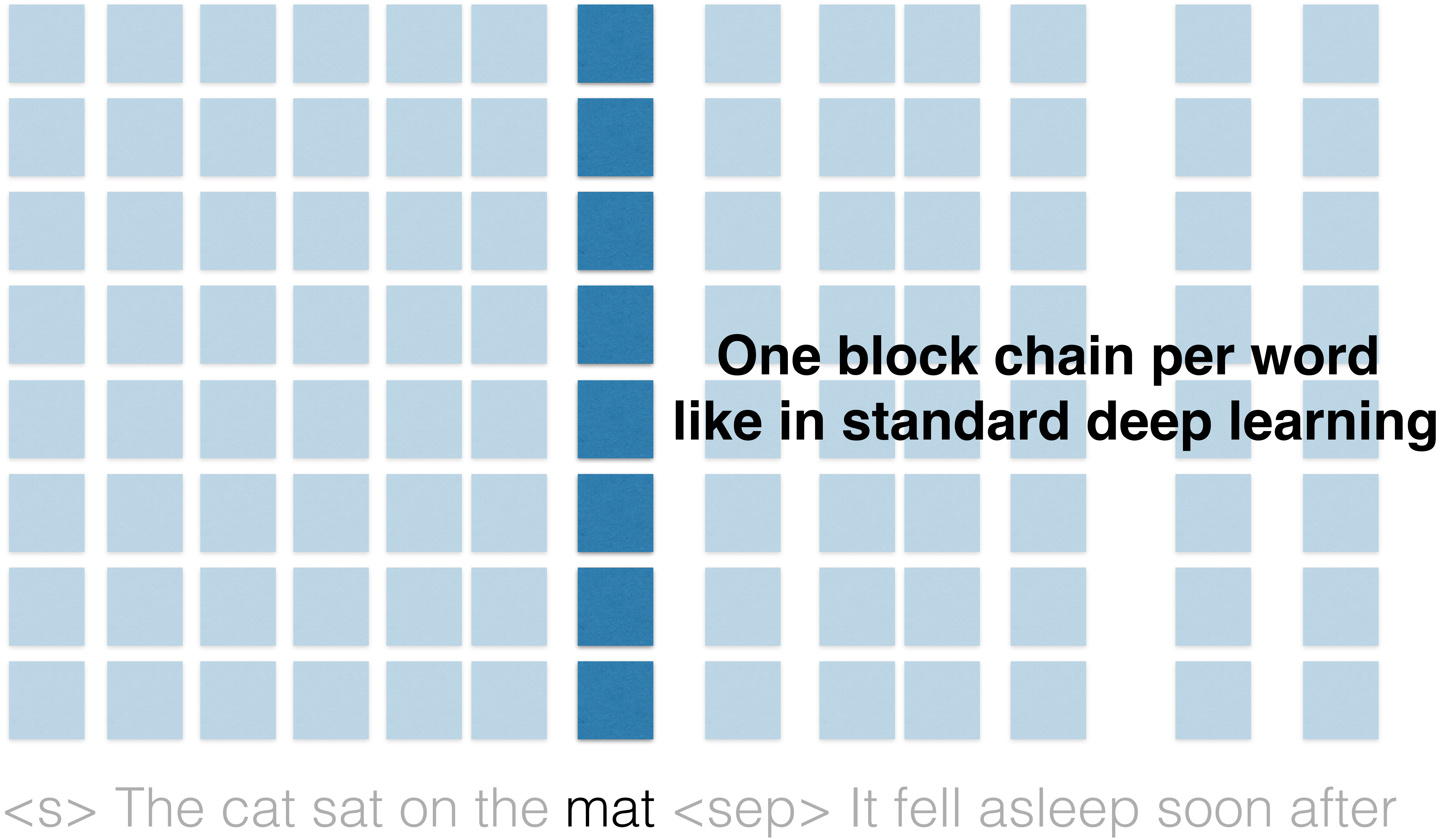
BERT



<s> The cat sat on the mat <sep> It fell asleep soon after

J. Devlin et al. “BERT: Pre-training of deep bidirectional transformers for language understanding”, arXiv:1810.04805, 2018

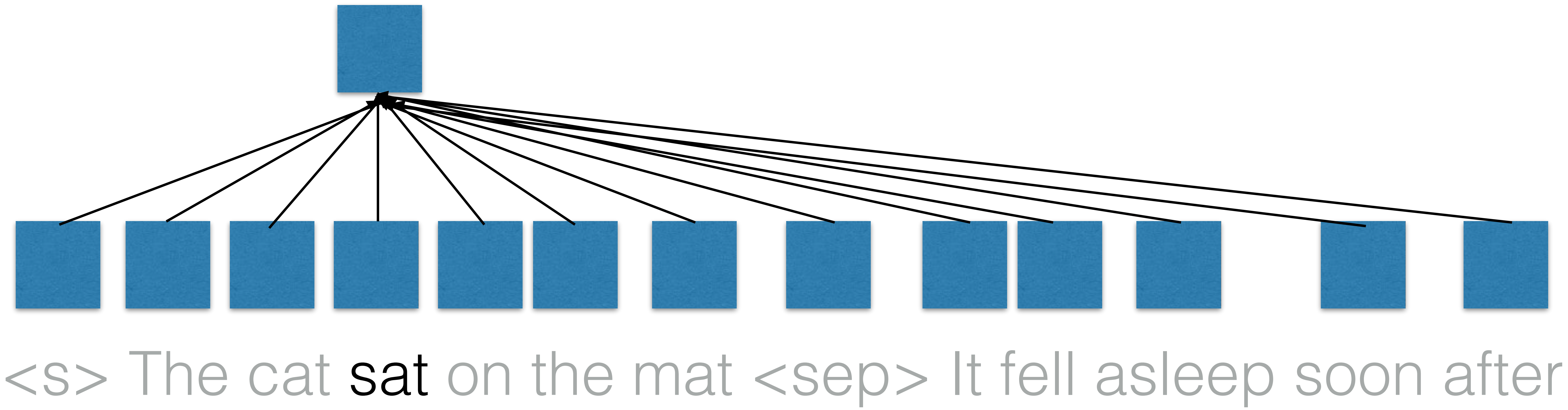
BERT



J. Devlin et al. “BERT: Pre-training of deep bidirectional transformers for language understanding”, arXiv:1810.04805, 2018

BERT

Each block receives input from all the blocks below.
Mapping must handle variable length sequences…



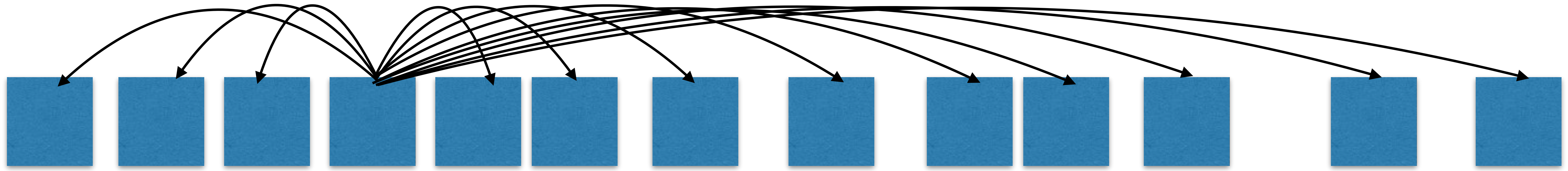
J. Devlin et al. "BERT: Pre-training of deep bidirectional transformers for language understanding", arXiv:1810.04805, 2018

BERT

This accomplished by using attention (each block is a Transformer)

For each layer and for each block in a layer do (simplified version):

1) let each current block representation at this layer be: h_j
2) compute dot products: $h_i \cdot h_j$
3) normalize scores: $\alpha_i = \frac{\exp(h_i \cdot h_j)}{\sum_k \exp(h_k \cdot h_j)}$
4) compute new block representation as in: $h_j \leftarrow \sum_k \alpha_k h_k$



<s> The cat sat on the mat <sep> It fell asleep soon after

A. Vaswani et al. "Attention is all you need", NIPS 2017

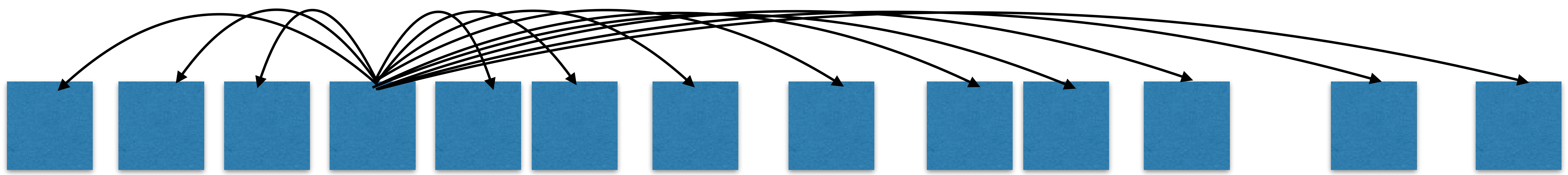
BERT

This accomplished by using attention
(each block is a Transformer)

For each layer and for each block in a layer do (simplified version):

1) let each current block representation at this layer be: h_j
2) compute dot products: $h_i \cdot h_j$
3) normalize scores: $\alpha_i = \frac{\exp(h_i \cdot h_j)}{\sum_k \exp(h_k \cdot h_j)}$
4) compute new block representation as in: $h_j \leftarrow \sum_k \alpha_k h_k$

in practice different features are used at each of these steps…

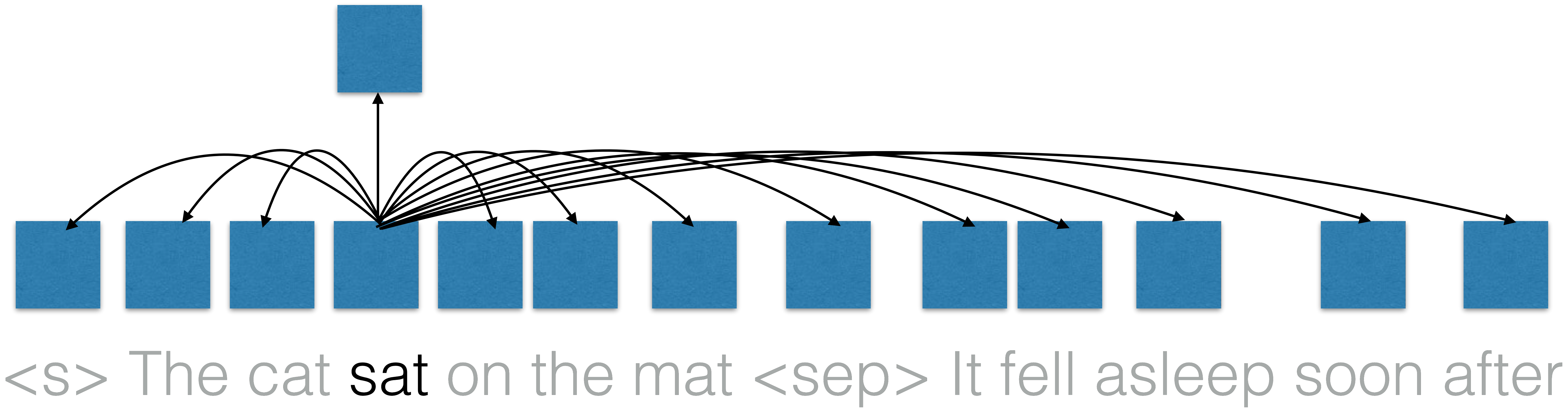


<s> The cat **sat** on the mat <sep> It fell asleep soon after

A. Vaswani et al. "Attention is all you need", NIPS 2017

BERT

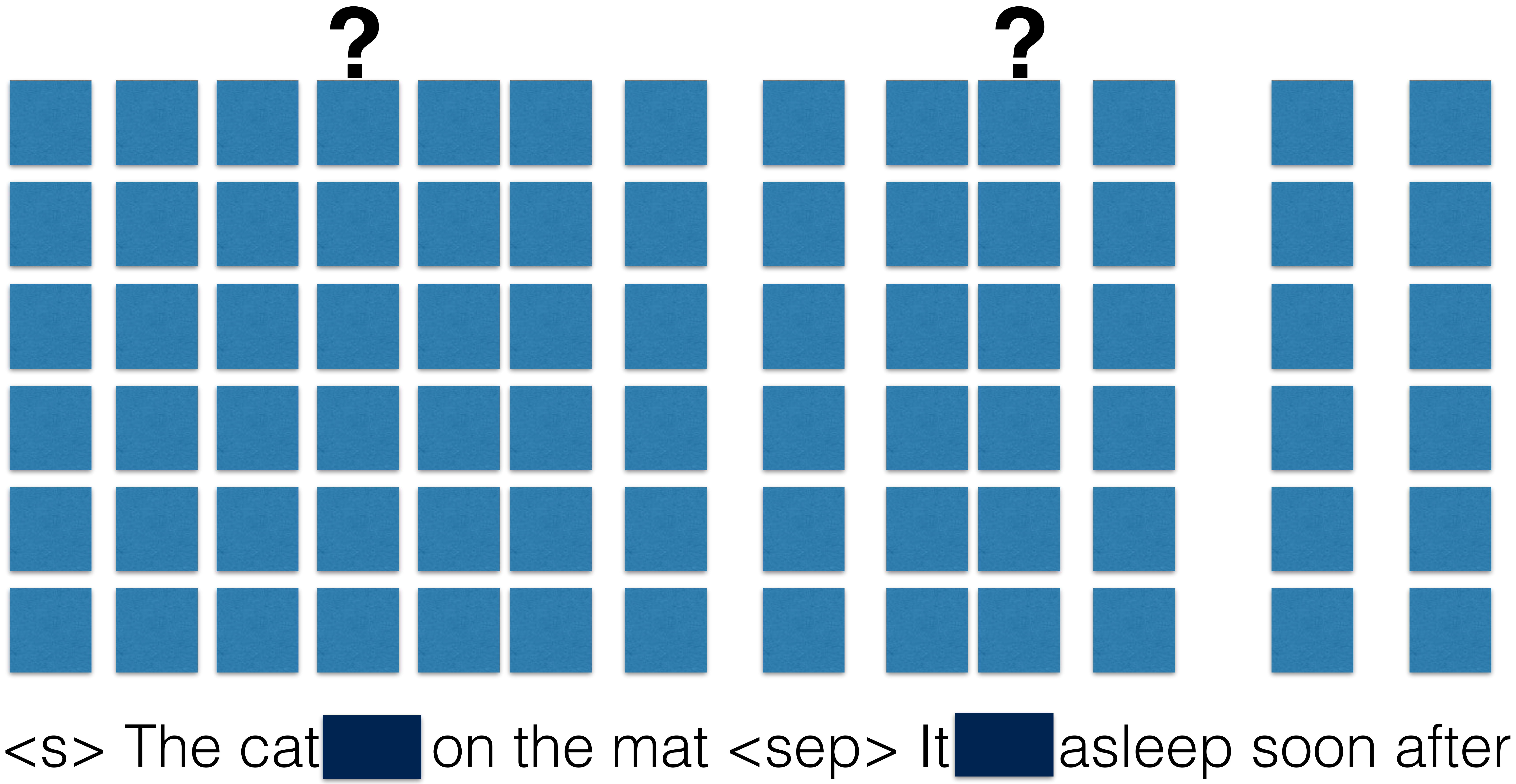
The representation of each word at each layer depends on all the words in the context. And there are lots of such layers…



J. Devlin et al. "BERT: Pre-training of deep bidirectional transformers for language understanding", arXiv:1810.04805, 2018

BERT: Training

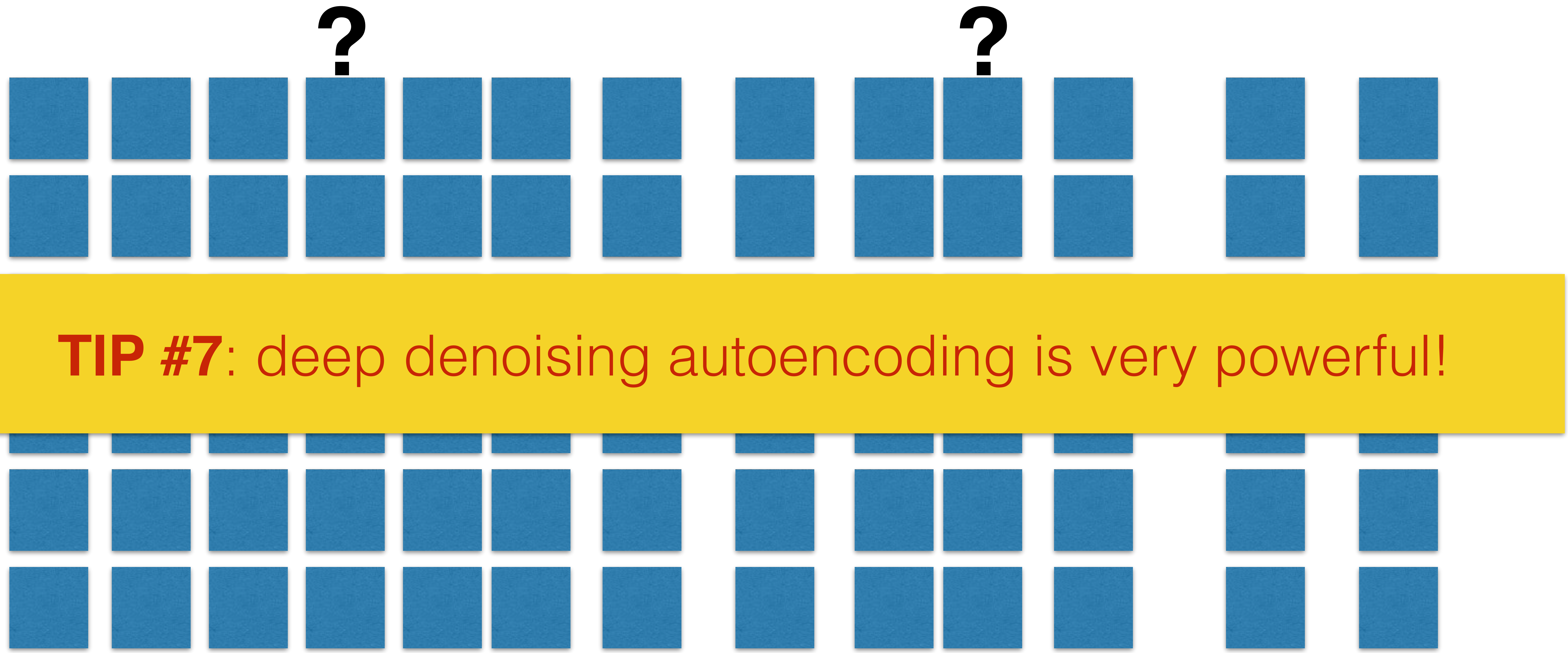
Predict blanked out words.



J. Devlin et al. “BERT: Pre-training of deep bidirectional transformers for language understanding”, arXiv:1810.04805, 2018

BERT: Training

Predict blanked out words.



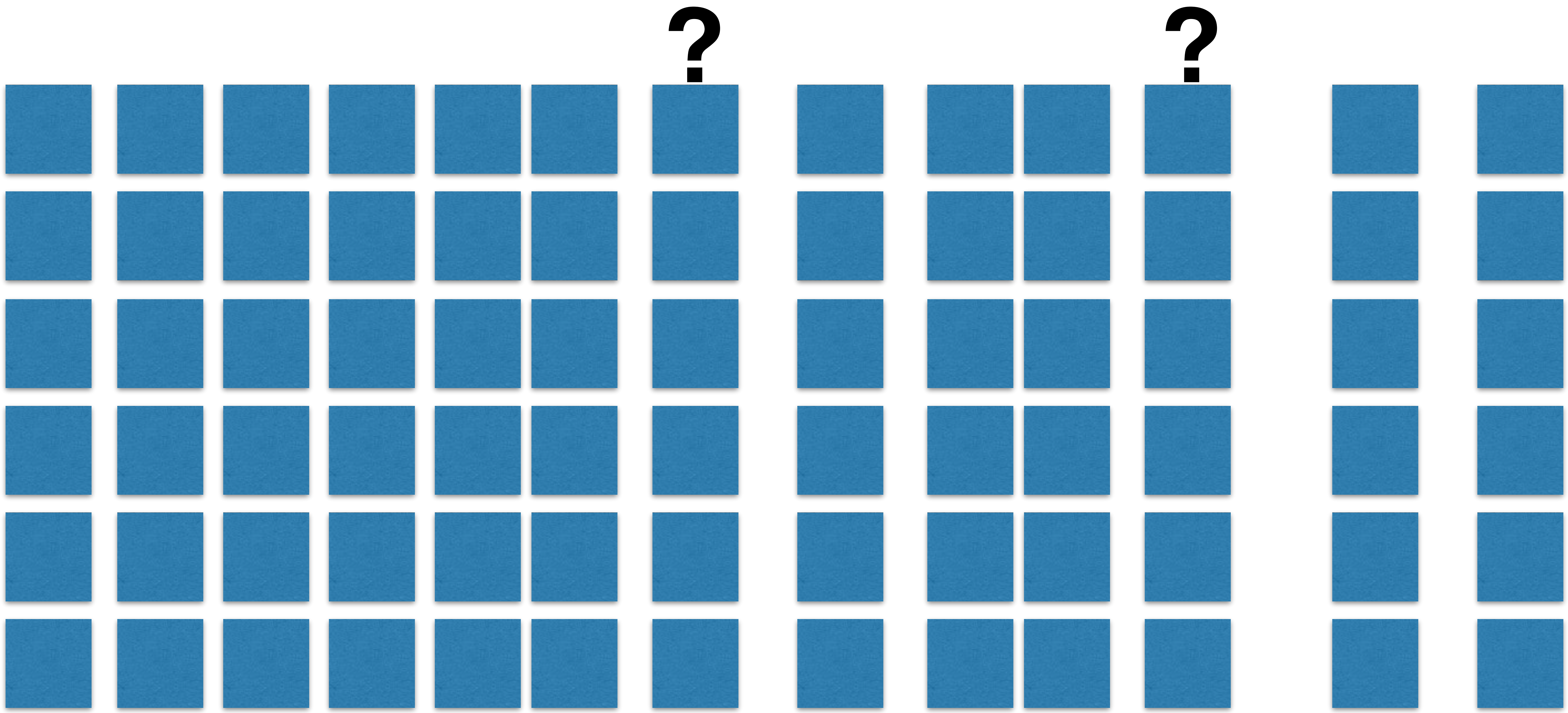
TIP #7: deep denoising autoencoding is very powerful!

<s> The cat [redacted] on the mat <sep> It [redacted] asleep soon after

J. Devlin et al. “BERT: Pre-training of deep bidirectional transformers for language understanding”, arXiv:1810.04805, 2018

BERT: Training

Predict words which were replaced with random words.

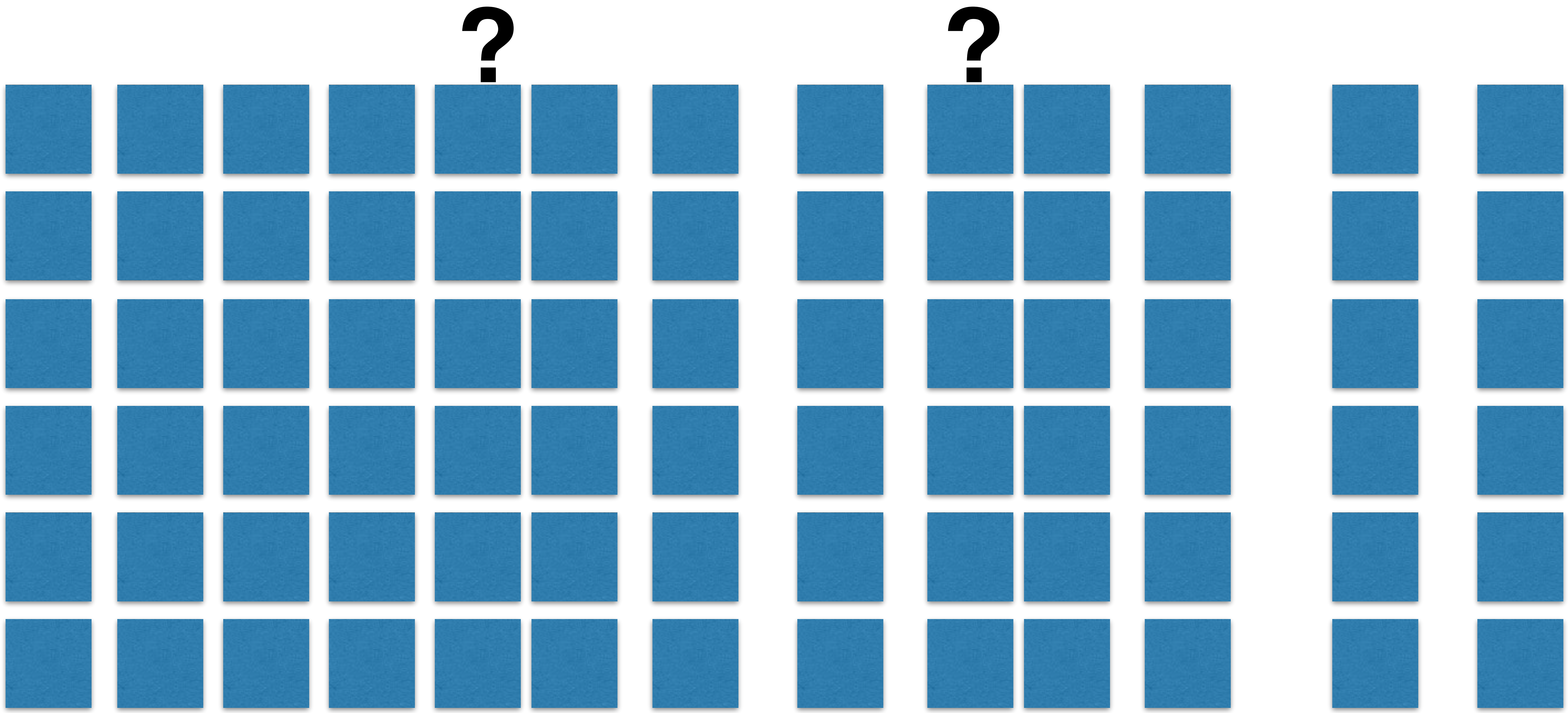


<s> The cat sat on the wine <sep> It fell scooter soon after

J. Devlin et al. “BERT: Pre-training of deep bidirectional transformers for language understanding”, arXiv:1810.04805, 2018

BERT: Training

Predict words from the input.

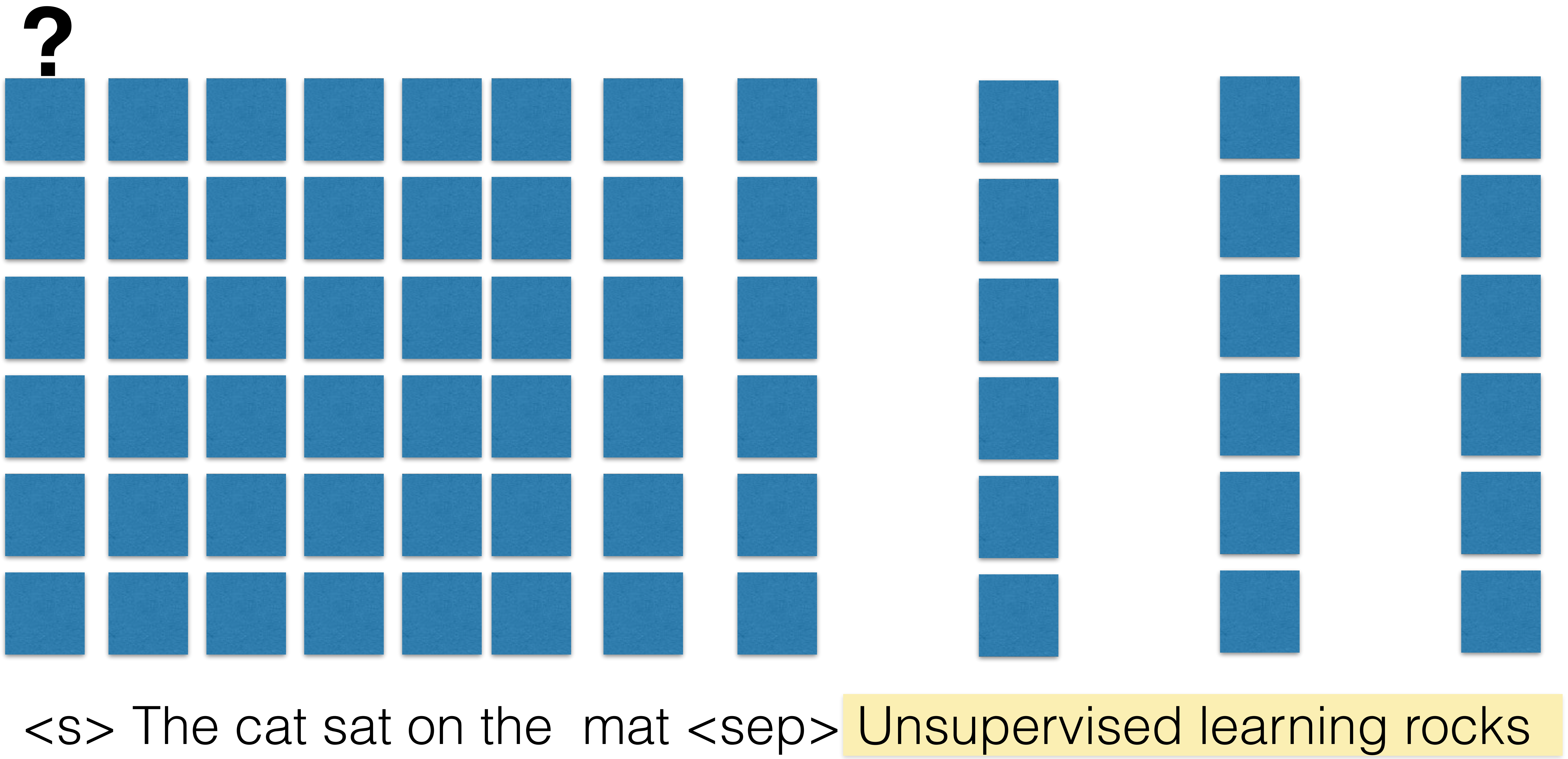


<s> The cat sat on the mat <sep> It fell asleep soon after

J. Devlin et al. "BERT: Pre-training of deep bidirectional transformers for language understanding", arXiv:1810.04805, 2018

BERT: Training

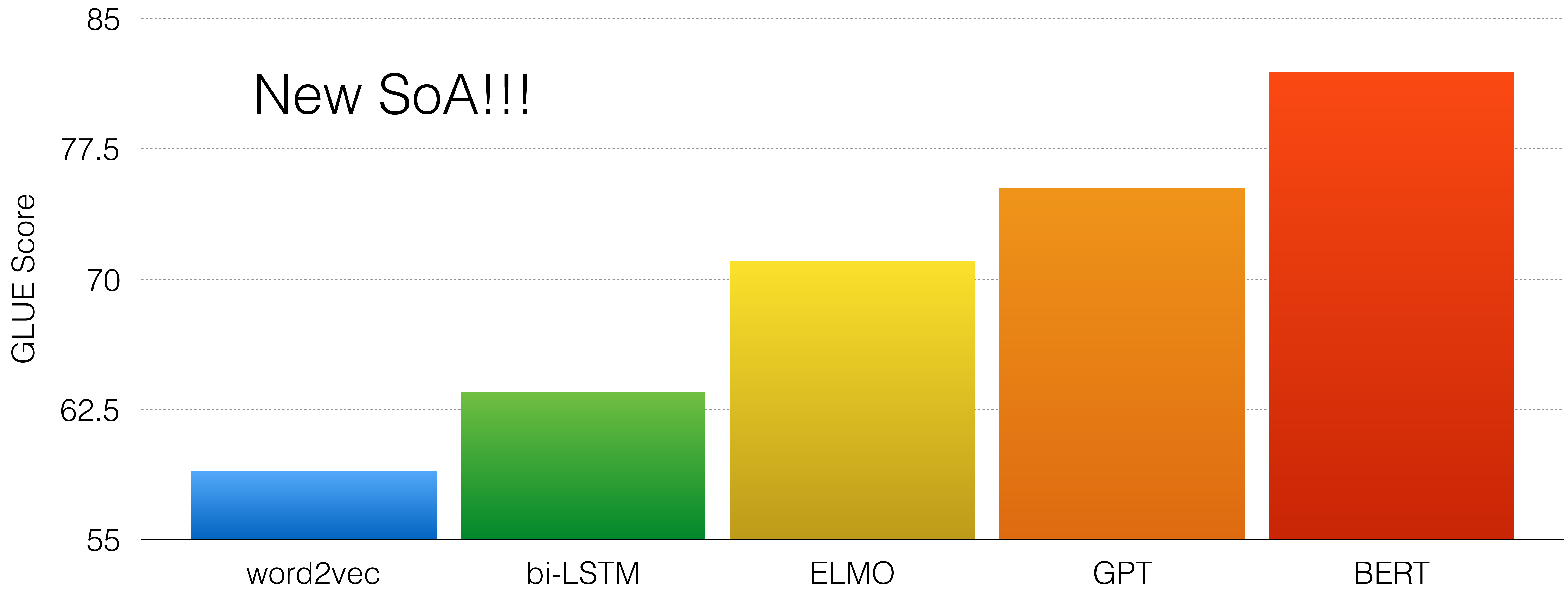
Predict whether the next sentence is taken at random.



J. Devlin et al. "BERT: Pre-training of deep bidirectional transformers for language understanding", arXiv:1810.04805, 2018

GLUE Benchmark (11 tasks)

Unsupervised pretraining followed by supervised finetuning



J. Devlin et al. “BERT: Pre-training of deep bidirectional transformers for language understanding”, arXiv:1810.04805, 2018

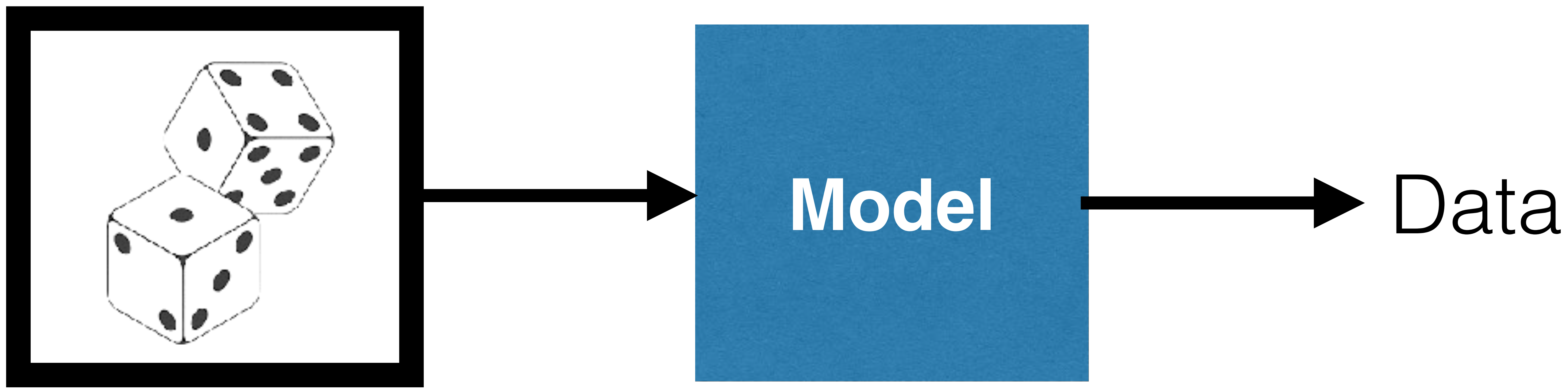
Conclusions on Learning Representation from Text

- Unsupervised learning has been very successful in NLP.
- Key idea: learn (deep) representations by predicting a word from the context (or vice versa).
- Current SoA performance across a large array of tasks.

Overview

- Practical Recipes of Unsupervised Learning
  - Learning representations
  - **Learning to generate samples (just a brief mention)**
  - Learning to map between two domains
- Open Research Problems

Generative Models



Useful for:

- learning representations (rarely the case nowadays),
- useful for planning (only in limited settings), or
- just for *fun* (most common use-case today)…

Generative Models: Vision

- GAN variants currently dominate the field.

T. Kerras et al. “Progressive growing of GANs for improved quality, stability, and variation”, ICLR 2018

Generative Models: Vision

- GAN variants currently dominate the field.

A. Brock et al. “Large scale GAN training for high fidelity natural image synthesis” arXiv 1809:11096 2018

Generative Models: Vision

- GAN variants currently dominate the field.
  A. Brock et al. “Large scale GAN training for high fidelity natural image synthesis” arXiv 1809:11096 2018

- Other approaches:
  - Auto-regressive
    A. Oord et al. “Conditional image generation with PixelCNN”, NIPS 2016
  - GLO
    P. Bojanowski et al. “Optimizing the latent state of generative networks”, ICML 2018
  - Flow-based algorithms.
    G. Papamakarios et al. “Masked auto-regressive flow for density estimation”, NIPS 2017

- Choice of architecture (CNN) seems more crucial than actual learning algorithm.

Generative Models: Vision

Open challenges:

- how to model high dimensional distributions,
- how to model uncertainty,
- meaningful metrics & evaluation tasks!

Anonymous “GenEval: A benchmark suite for evaluating generative models”, in submission to ICLR 2019

Generative Models: Text

- Auto-regressive models (RNN/CNN/Transformers) are good at generating short sentences. See Alex's examples.

  I. Serban et al. "Building end-to-end dialogue systems using generative hierarchical neural network models" AAAI 2016

- Retrieval-based approaches are often used in practice.

  A. Bordes et al. "Question answering with subgraph embeddings" EMNLP 2014
  R. Yan et al. "Learning to Respond with Deep Neural Networks for Retrieval-Based Human-Computer Conversation System", SIGIR 2016
  M. Henderson et al. "Efficient natural language suggestion for smart reply", arXiv 2017
  …

- The two can be combined

  J. Gu et al. "Search Engine Guided Non-Parametric Neural Machine Translation", arXiv 2017
  K. Guu et al. "Generating Sentences by Editing Prototypes", ACL 2018
  …

Generative Models: Text

Open challenges:

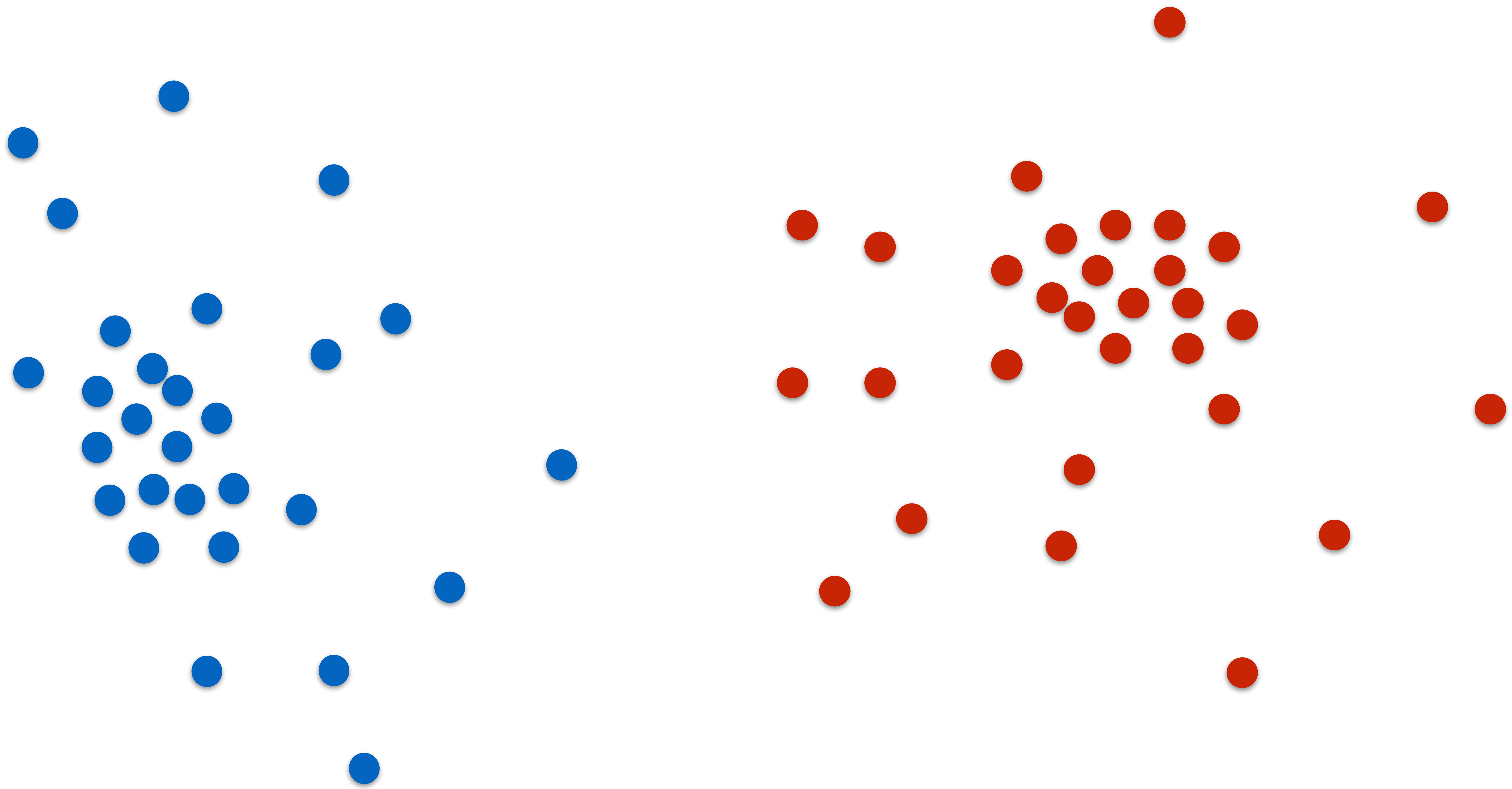
- how to generate documents (long pieces of text) that are coherent,
- how to keep track of state,
- how to model uncertainty,
  M. Ott et al. “Analyzing uncertainty in NMT” ICML 2018
- how to ground,
  starting with D. Roy / J. Siskind’s work from early 2000’s
- meaningful metrics & standardized tasks!

Overview

- Practical Recipes of Unsupervised Learning
  - Learning representations
  - Learning to generate samples
  - **Learning to map between two domains**
- Open Research Problems

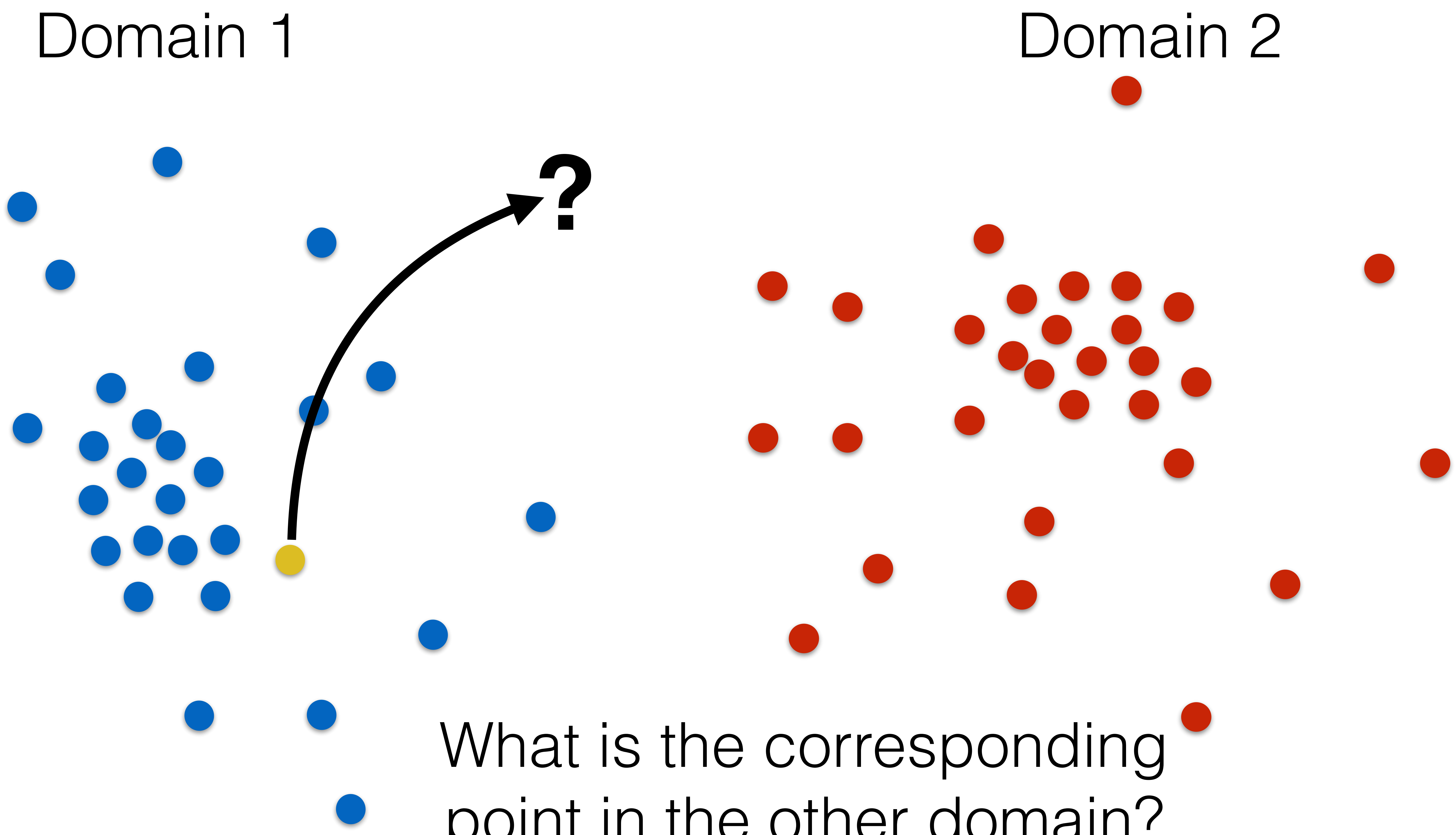
Learning to Map

Domain 1 Domain 2



Toy illustration of the data

Learning to Map



What is the corresponding point in the other domain?

Toy illustration of the data

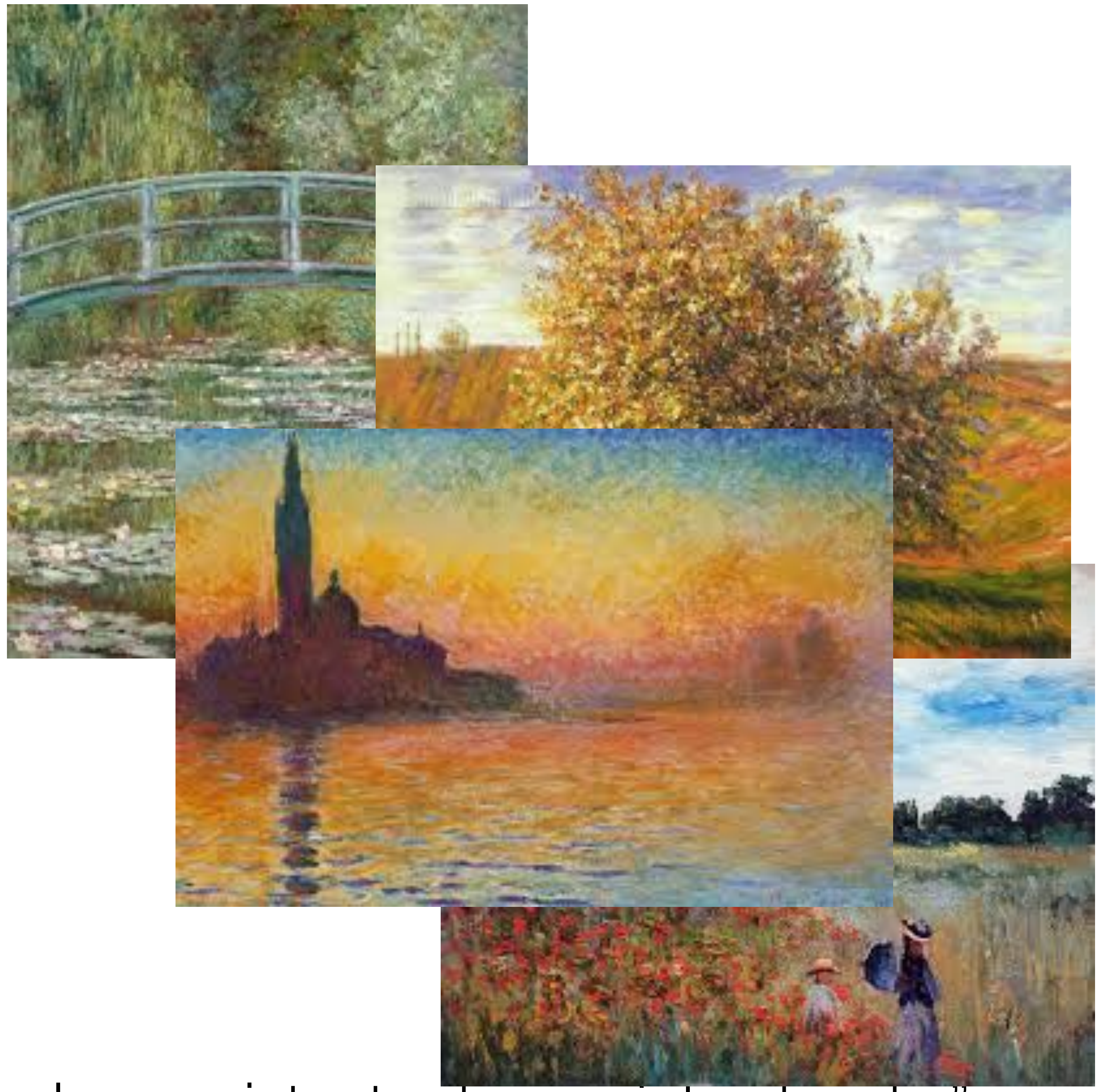
Why Learning to Map

- There are fun applications: making analogies in vision.
- It is useful; e.g., enables to leverage lots of (unlabeled) monolingual data in machine translation.
- Arguably, an AI agent has to be able to perform analogies to quickly adapt to a new environment.

Vision: Cycle-GAN

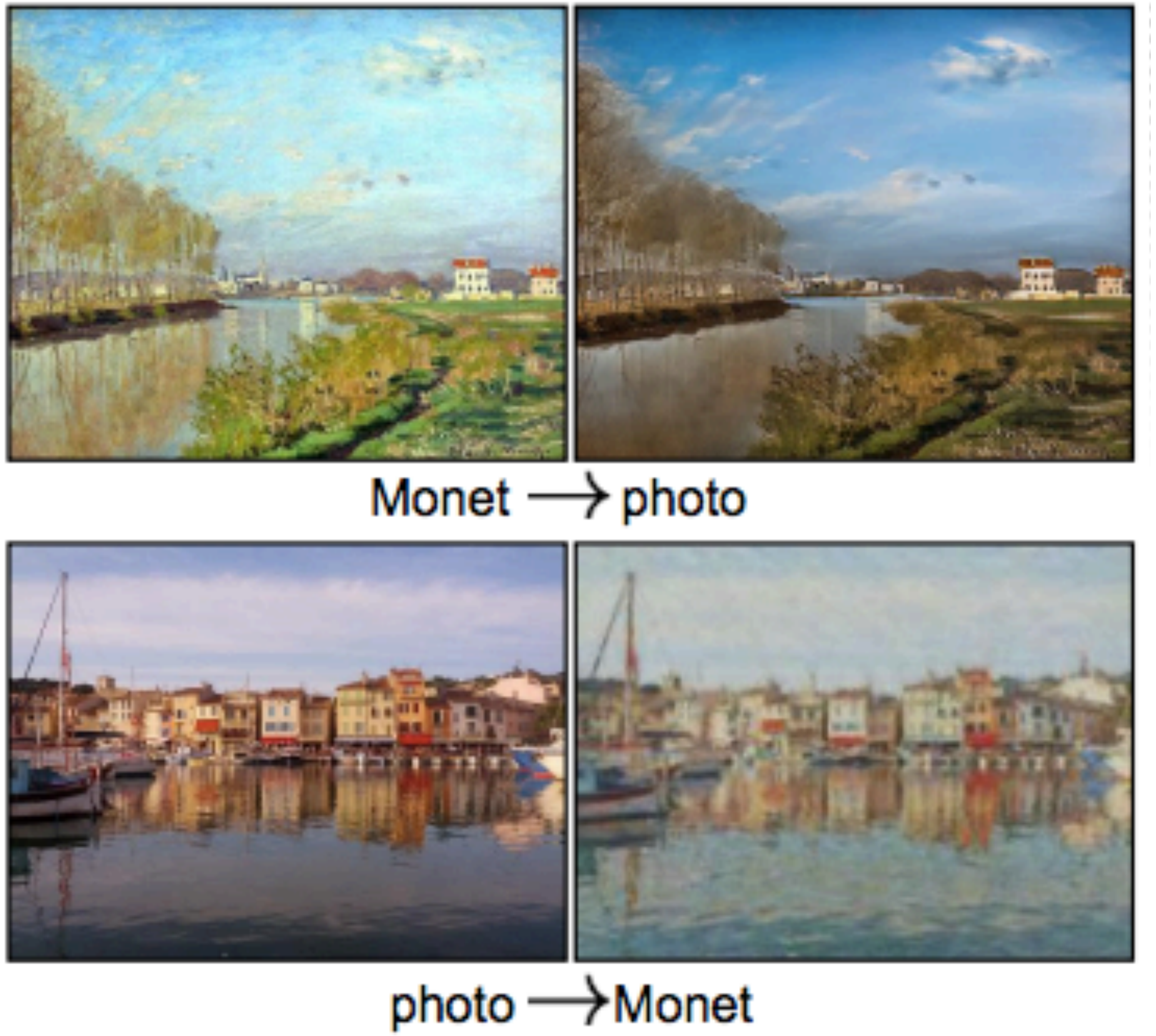
Domain 1

Domain 2



J. Zhu et al. “Unpaired image-to-image translation using cycle consistent adversarial networks”, ICCV 2017

Vision: Cycle-GAN

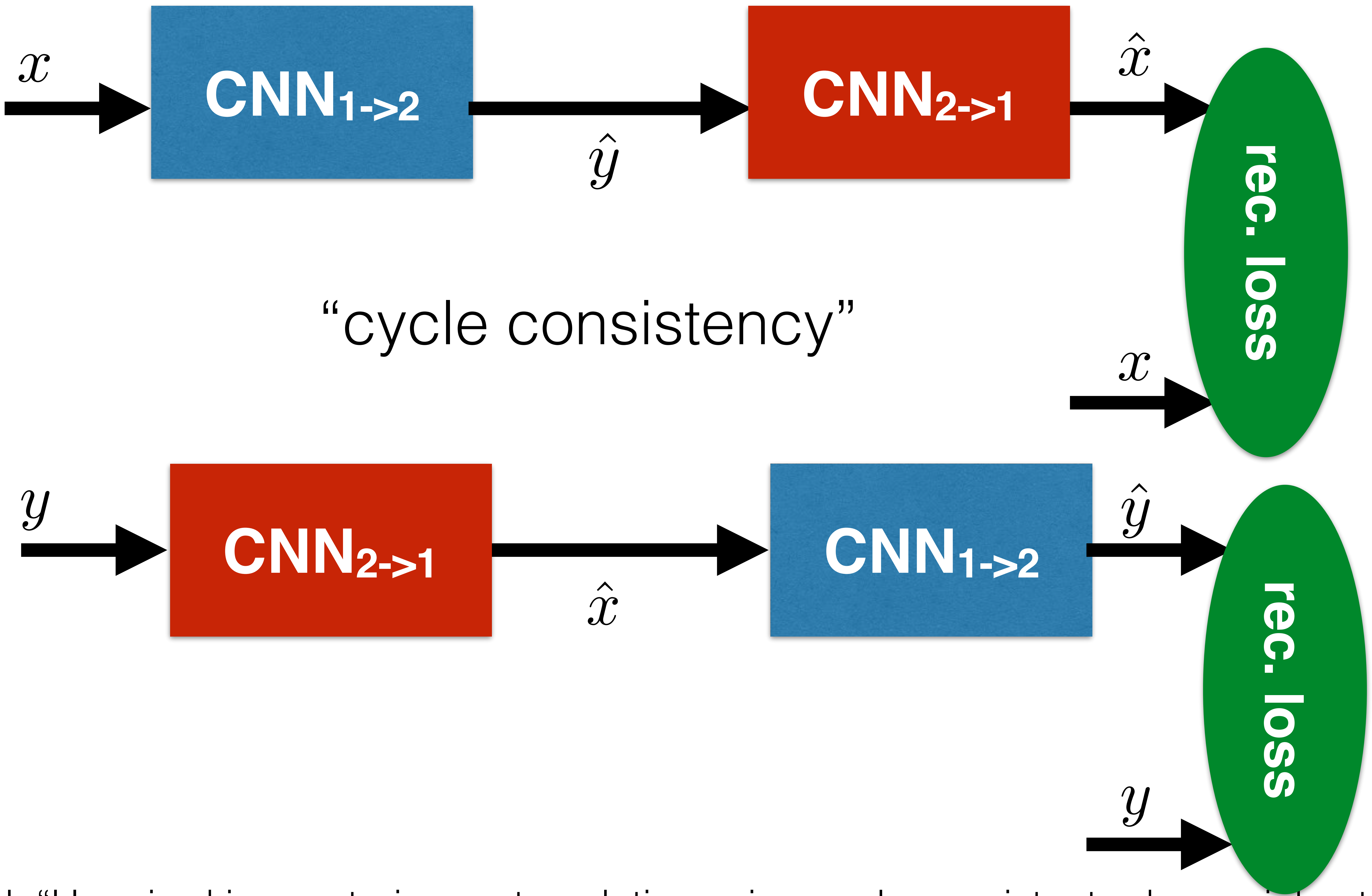


J. Zhu et al. “Unpaired image-to-image translation using cycle consistent adversarial networks”, ICCV 2017

Vision: Cycle-GAN

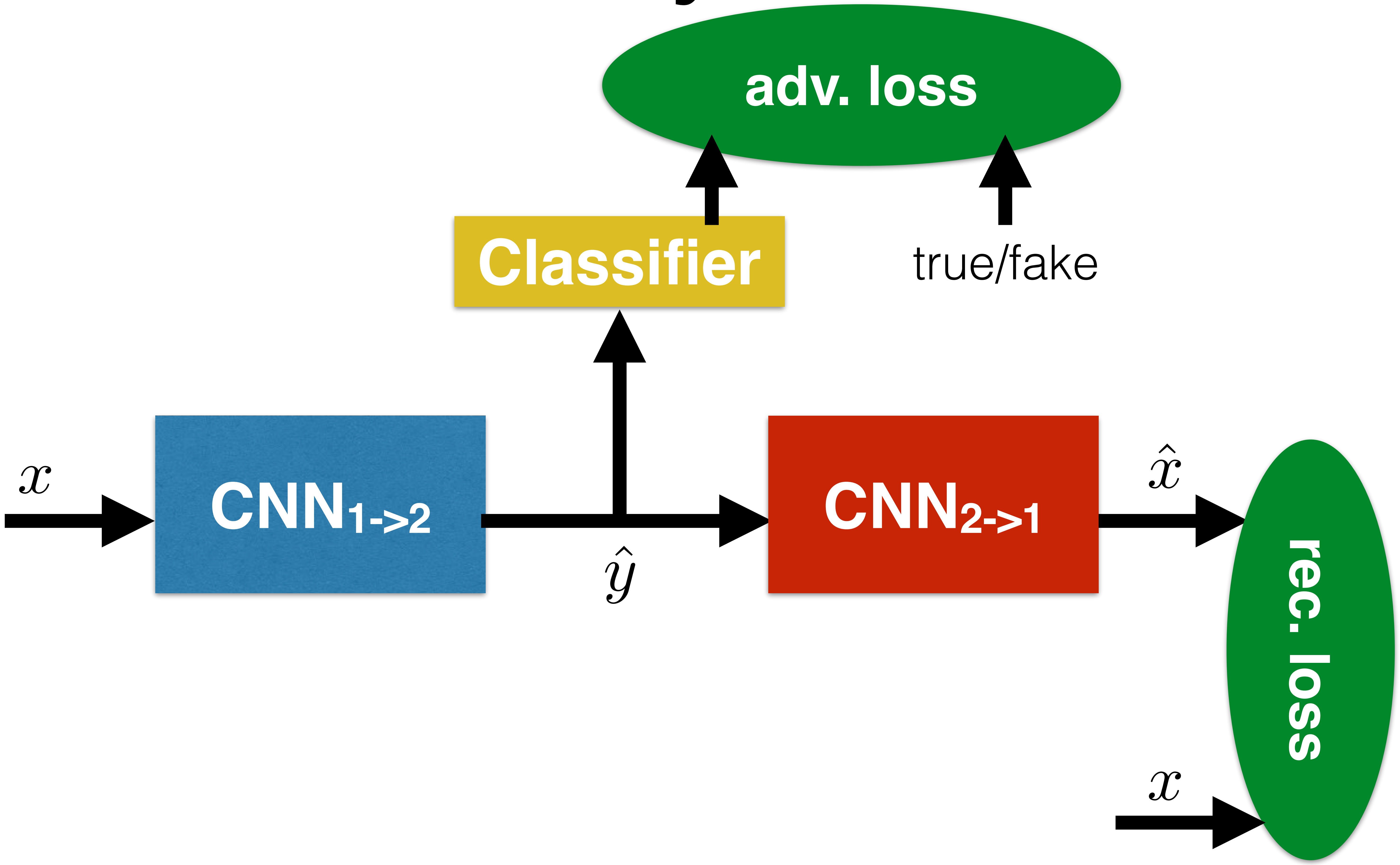
J. Zhu et al. “Unpaired image-to-image translation using cycle consistent adversarial networks”, ICCV 2017

Vision: Cycle-GAN



J. Zhu et al. “Unpaired image-to-image translation using cycle consistent adversarial networks”, ICCV 2017

Vision: Cycle-GAN



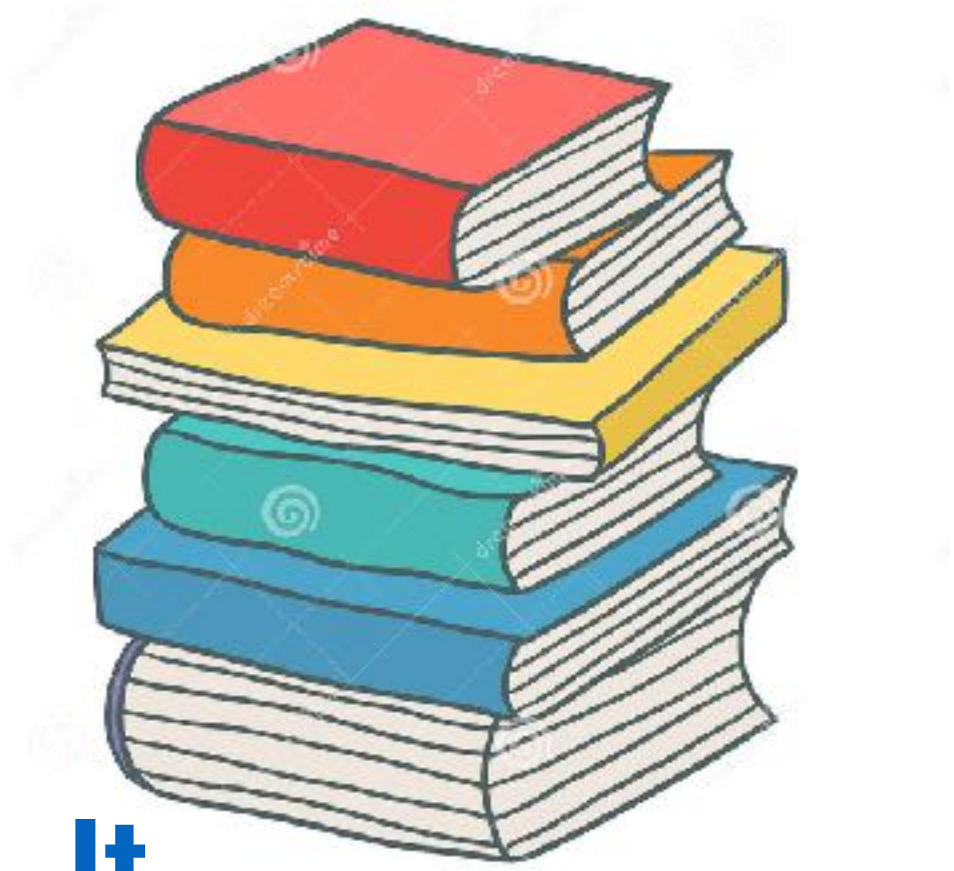
constrain generation to belong to desired domain

J. Zhu et al. “Unpaired image-to-image translation using cycle consistent adversarial networks”, ICCV 2017

Unsupervised Machine Translation

- Similar principles may apply also to NLP, e.g. for machine translation (MT).

En



It

Learning to translate without access to any single translation, just lots of (monolingual) data in each language.

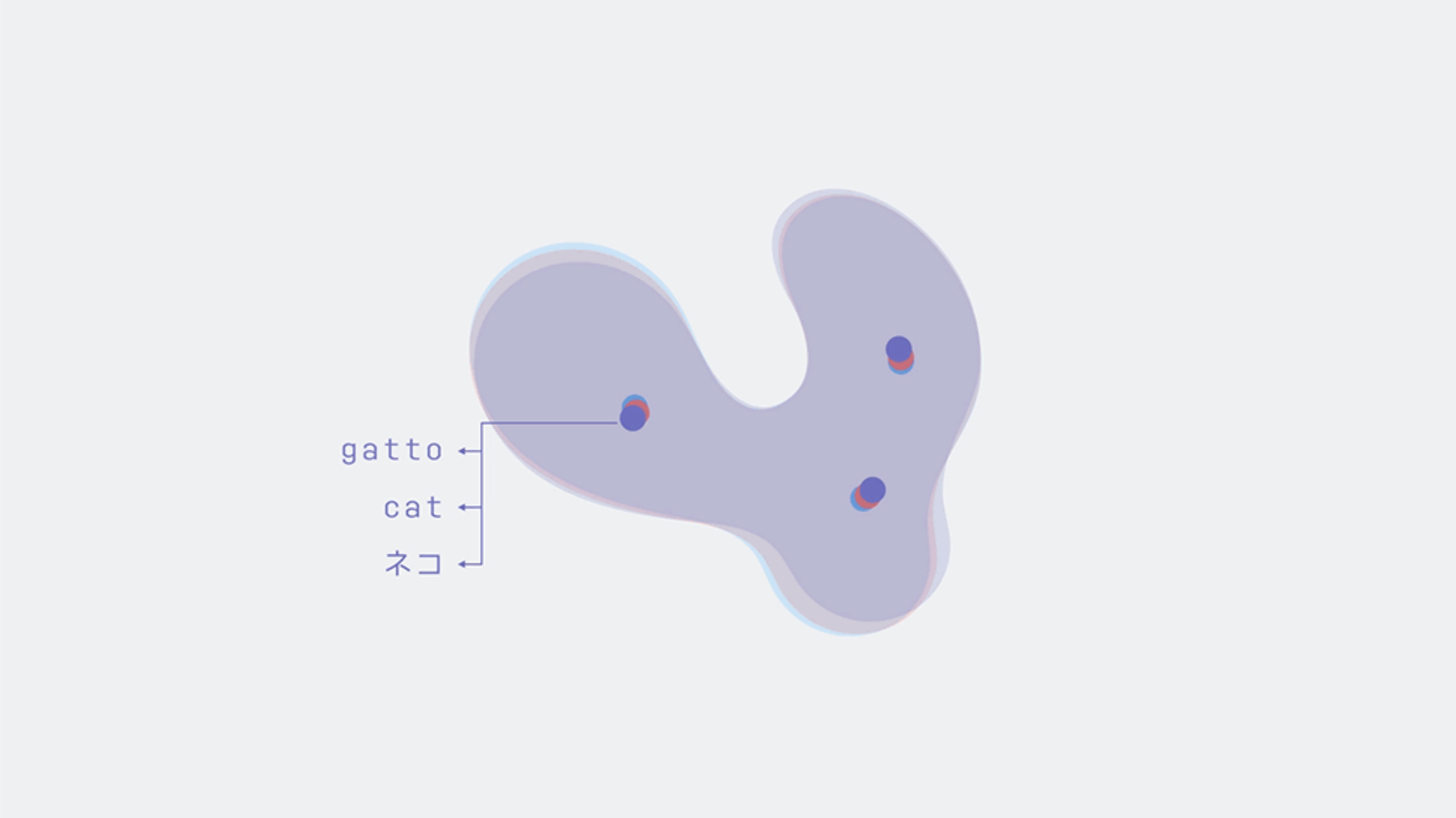
Unsupervised Machine Translation

- Similar principles may apply also to NLP for machine translation (MT).
- Can we do unsupervised MT?
  - There is little if any parallel data in most language pairs.
- Challenges:
  - discrete nature of text
  - domain mismatch
  - languages may have very different morphology, grammar, ..

Unsupervised **Word** Translation

- Motivation: A pre-requisite for unsupervised sentence translation.
- Problem: given two monolingual corpora in two different languages, estimate bilingual lexicon.
- Hint: the context of a word, is often similar across languages since each language refers to the same underlying physical world.

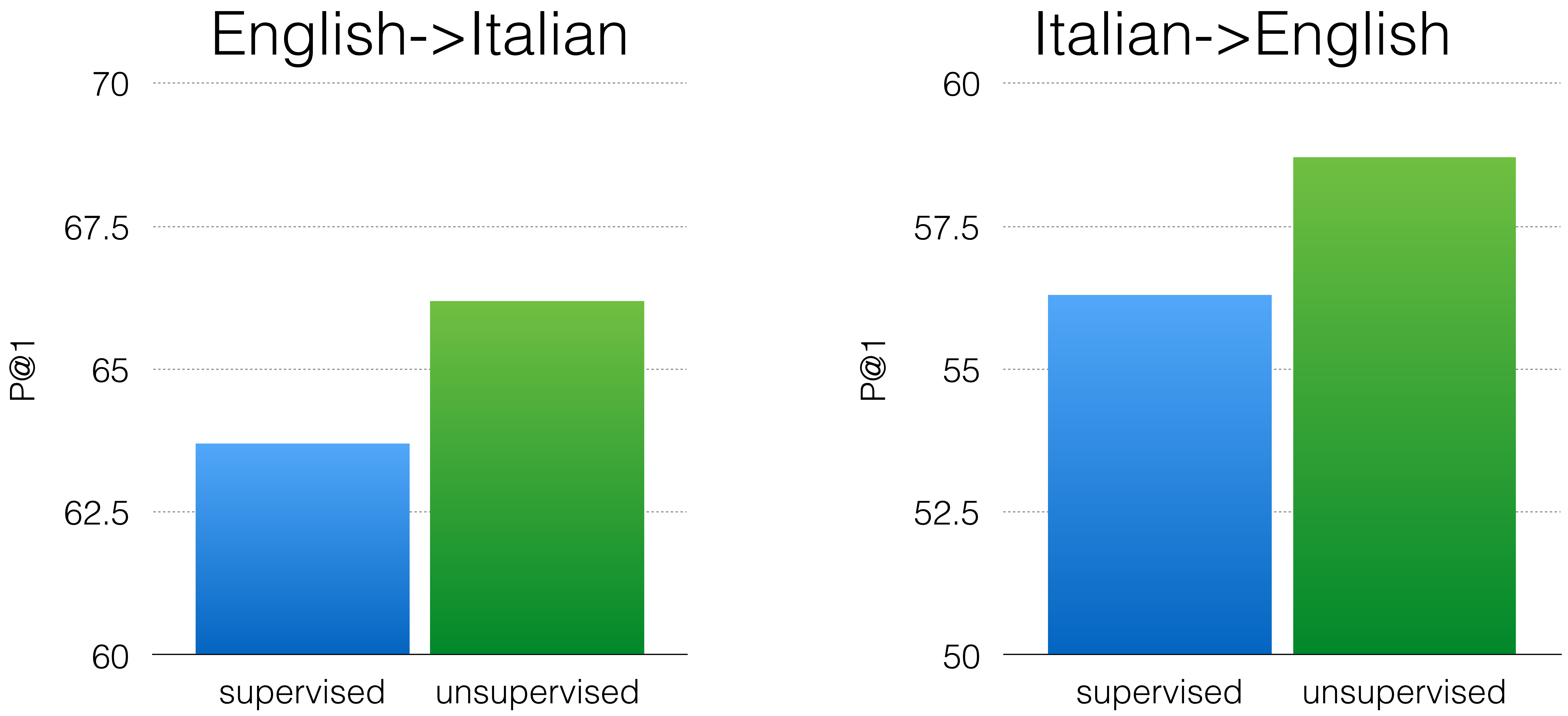
Unsupervised **Word** Translation



1) Learn embeddings separately.
2) Learn joint space via adversarial training + refinement.

A. Conneau et al. “Word translation without parallel data” ICLR 2018

Results on Word Translation



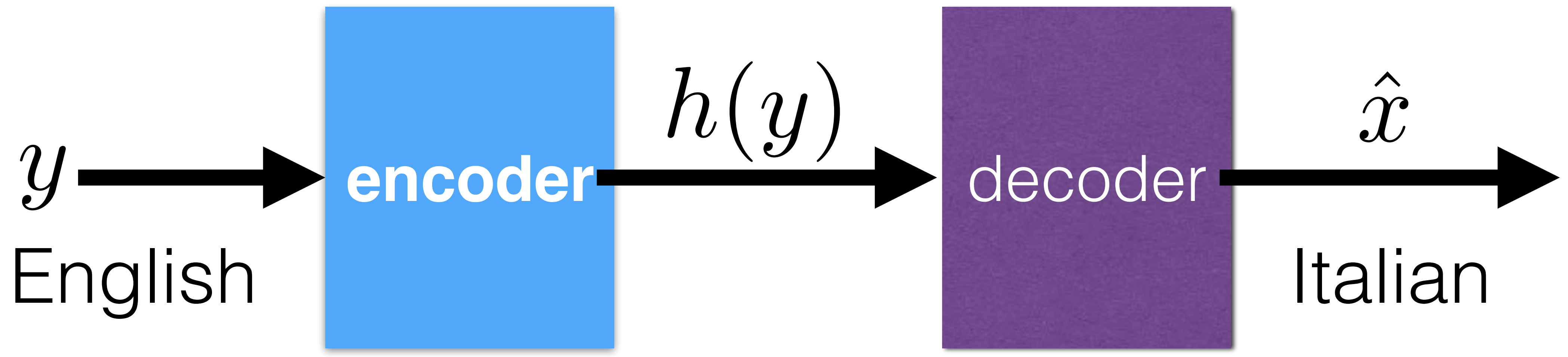
By using more anchor points and lots of unlabeled data, MUSE outperforms supervised approaches!

https://github.com/facebookresearch/MUSE

Naïve Application of MUSE

- In general, this may not work on sentences because:
    - Without leveraging compositional structure, space is exponentially large.
    - Need good sentence representations.
    - Unlikely that a linear mapping is sufficient to align sentence representations of two languages.

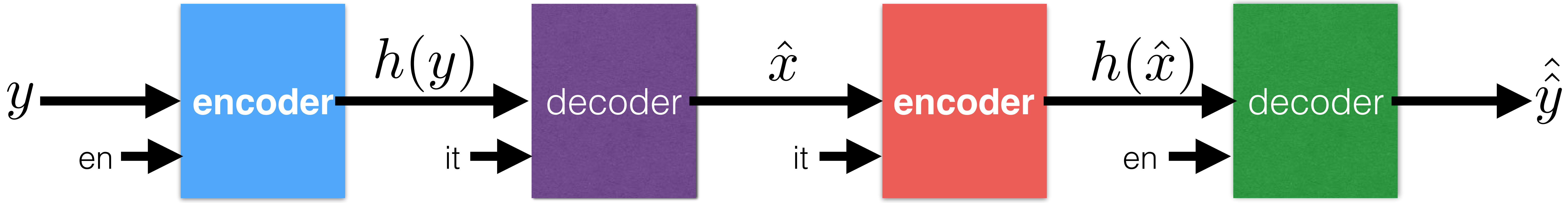
Method



We want to learn to translate, but we do not have targets…

G. Lample et al. "Phrase-based and neural unsupervised machine translation" EMNLP 2018

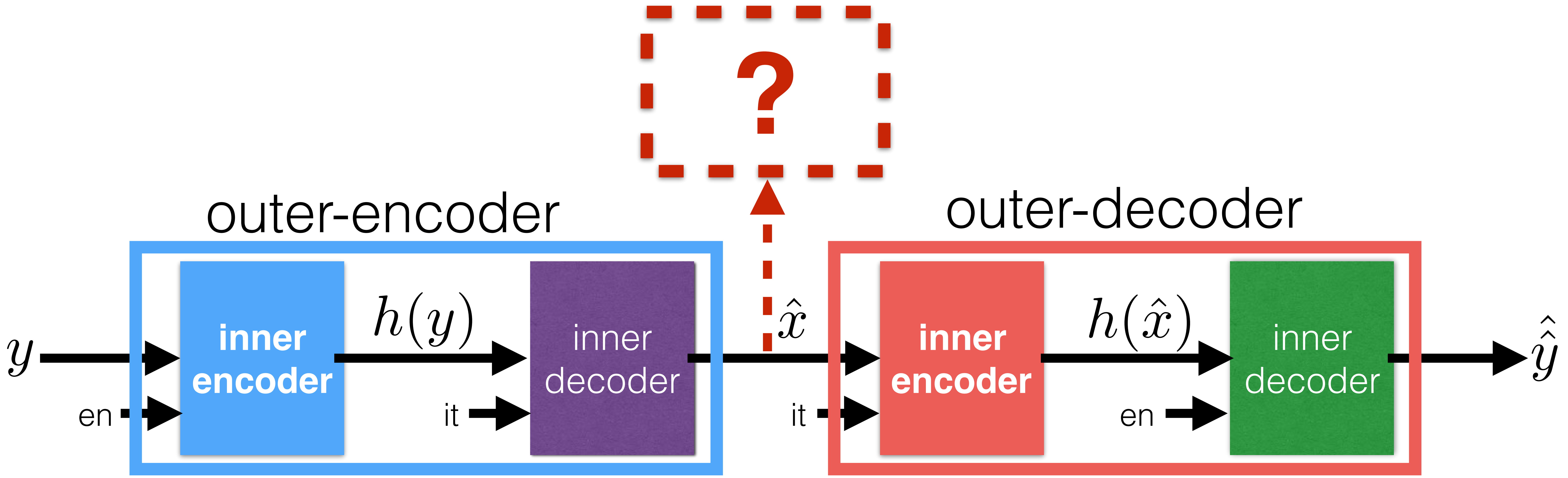
Method



use the same cycle-consistency principle (back-translation)

G. Lample et al. "Phrase-based and neural unsupervised machine translation" EMNLP 2018

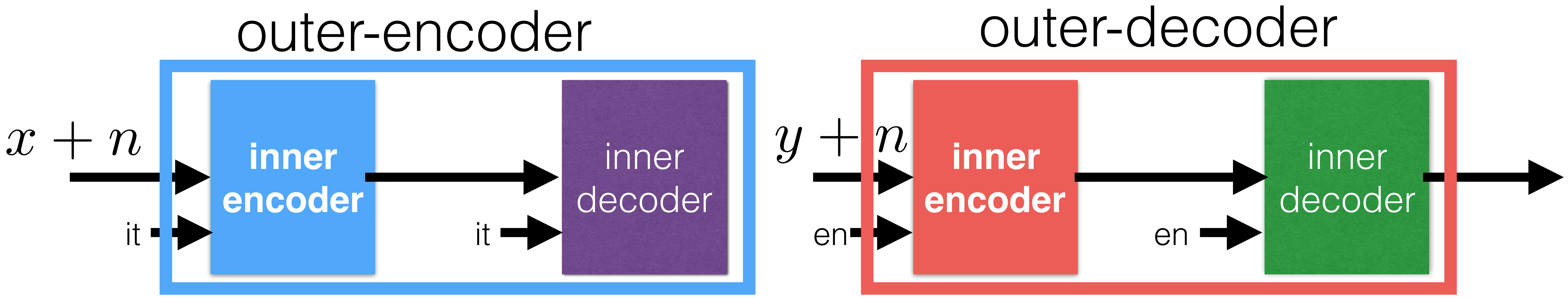
Method



How to ensure the intermediate output is a valid sentence?
Can we avoid back-propping through a discrete sequence?

G. Lample et al. "Phrase-based and neural unsupervised machine translation" EMNLP 2018

Adding Language Modeling

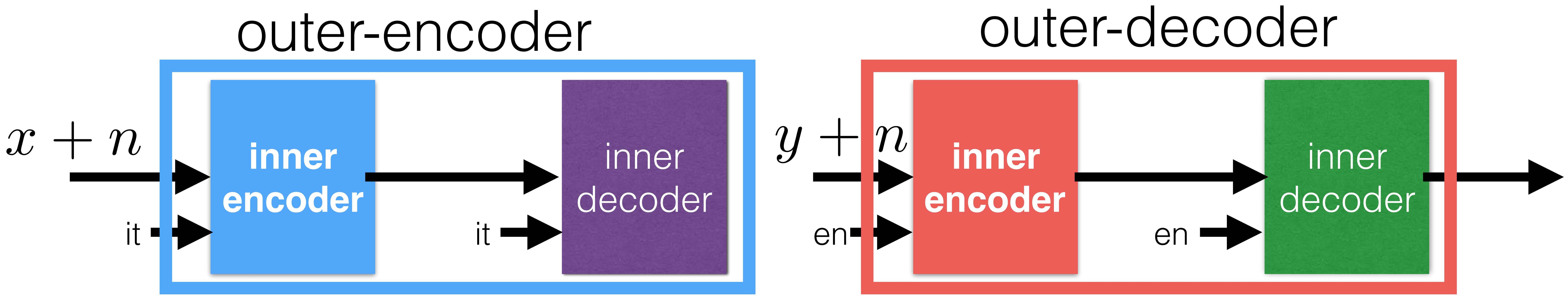


Since inner decoders are shared between the LM and MT task, it should constrain the intermediate sentence to be fluent.

Noise: word drop & swap.

G. Lample et al. "Phrase-based and neural unsupervised machine translation" EMNLP 2018

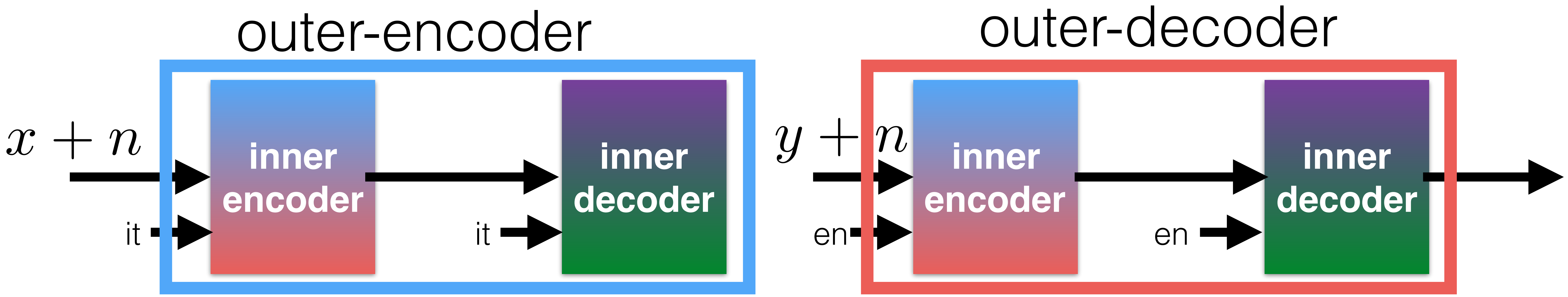
Adding Language Modeling



Potential issue: Model can learn to denoise well, reconstruct well from back-translated data and yet not translate well, if it splits the latent representation space.

G. Lample et al. "Phrase-based and neural unsupervised machine translation" EMNLP 2018

NMT: Sharing Latent Space

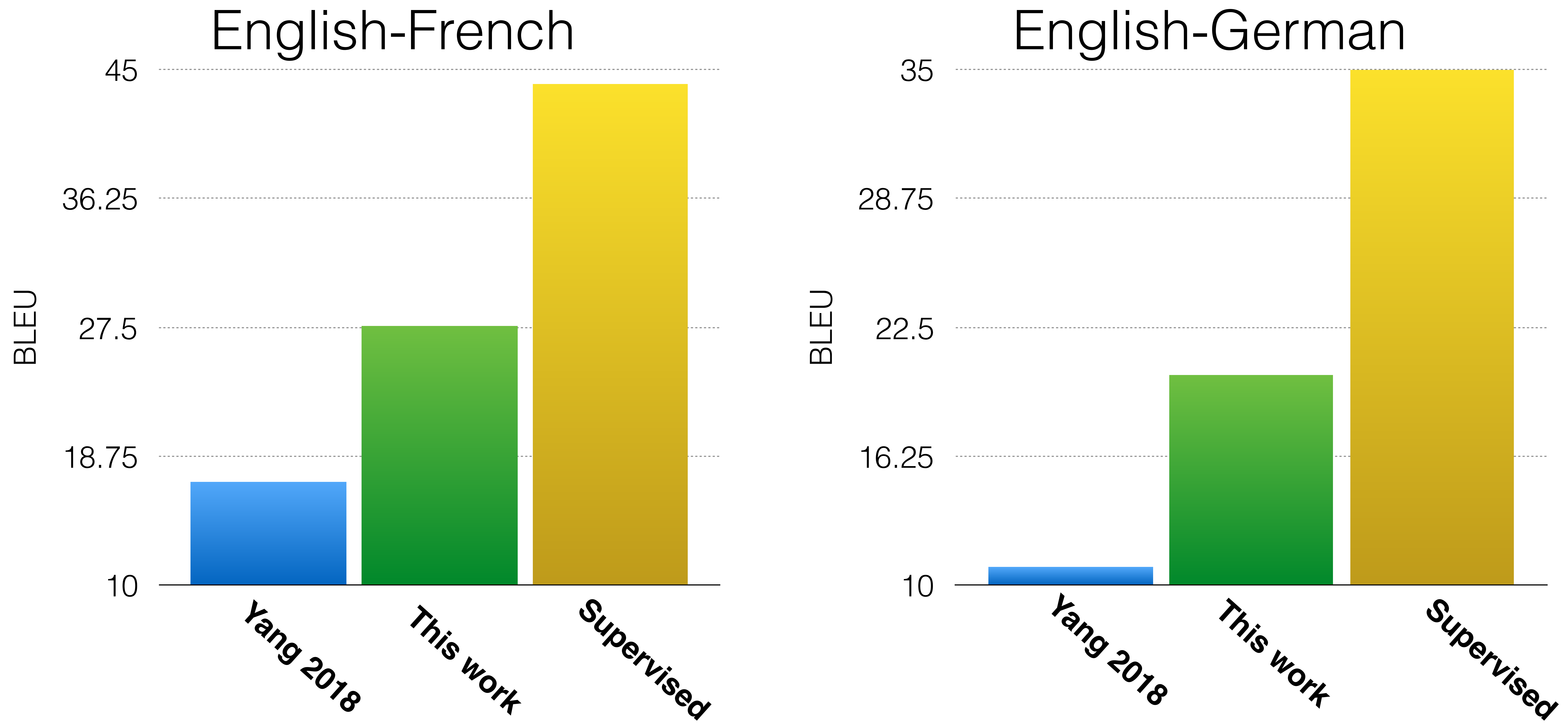


Sharing achieved via:

1) shared encoder (and also decoder).
2) joint BPE embedding learning / initialize embeddings with MUSE.

Note: first decoder token specifies the language on the target-side.

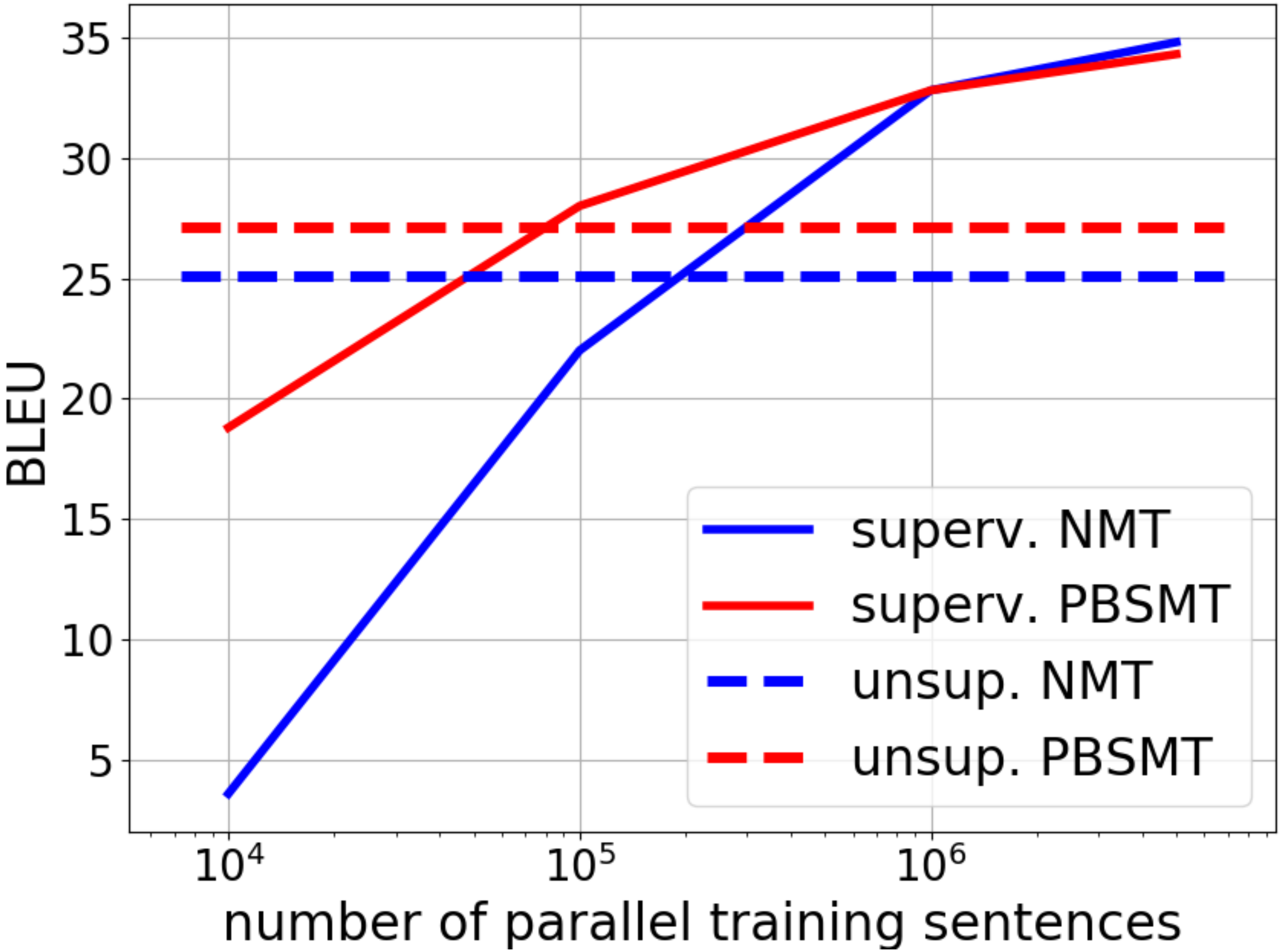
Experiments on WMT



Before 2018, performance of fully unsupervised methods was essentially 0 on these large scale benchmarks!

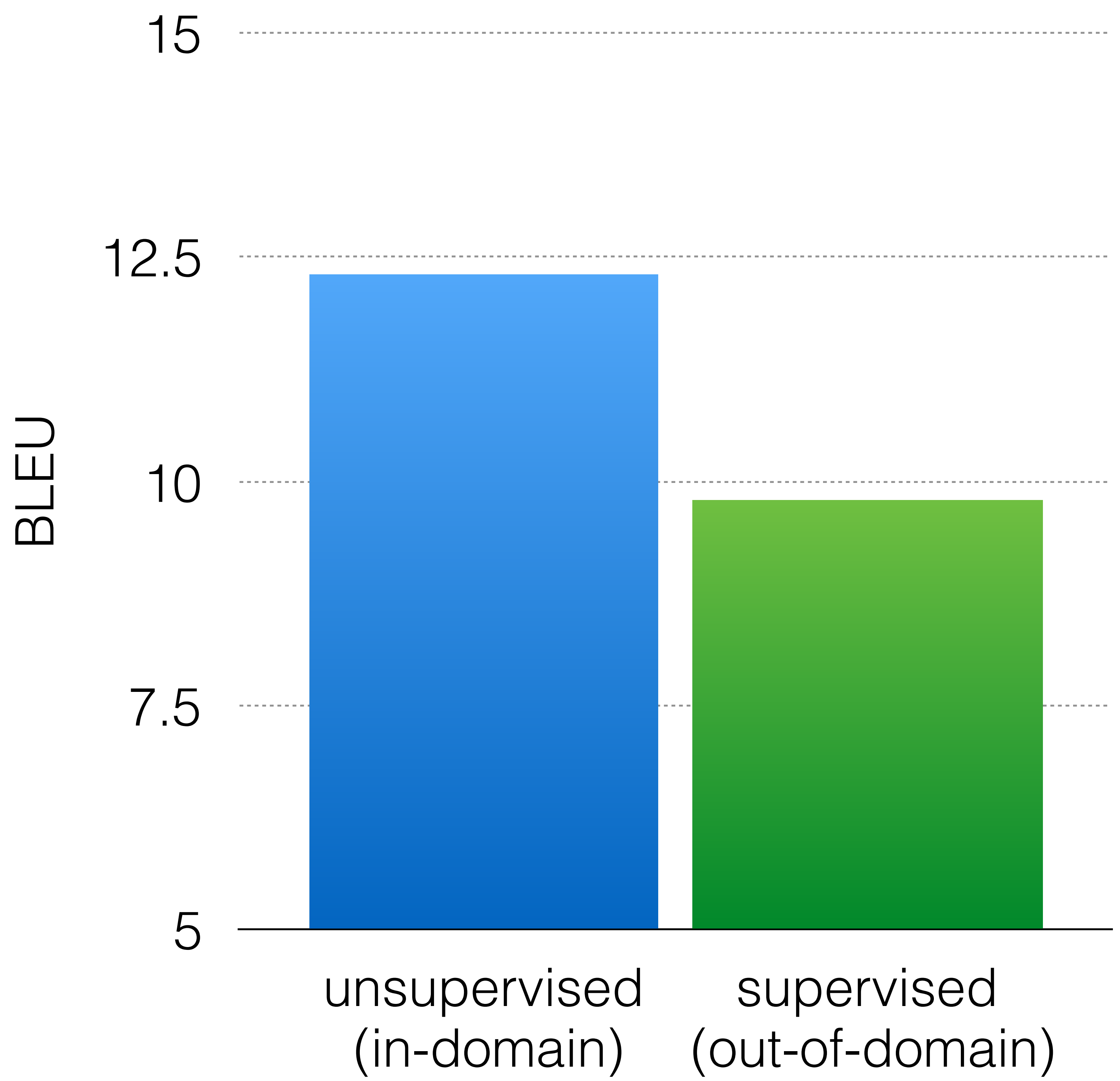
G. Lample et al. “Phrase-based and neural unsupervised machine translation” EMNLP 2018

Experiments on WMT



Distant & Low-Resource Language Pair: En-Ur

https://www.bbc.com/urdu/pakistan-44867259



G. Lample et al. "Phrase-based and neural unsupervised machine translation" EMNLP 2018

Conclusion on Unsupervised Learning to Translate

- General principles: initialization, matching target domain and cycle-consistency.
- Extensions: semi-supervised, more than two domains, more than a single attribute, …
- Challenges:
    - domain mismatch / ambiguous mappings
    - domains with very different properties

Overview

- Practical Recipes of Unsupervised Learning
  - Learning representations
  - Learning to generate samples (just a brief mention)
  - Learning to map between two domains
- **Open Research Problems**

Challenge #1: Metrics & Tasks

Unsupervised Feature Learning:
Q: What are good down-stream tasks?
What are good metrics for such tasks?

In NLP there is some consensus for this: **https://github.com/facebookresearch/SentEval**
https://gluebenchmark.com/

Generation:
Q: What is a good metric?

In NLP there has been some effort towards this: **http://www.statmt.org/**
http://www.parl.ai/

Challenge #1: Metrics & Tasks

Unsupervised Feature Learning:

Q: What are good down-stream tasks?
What are good metrics for such tasks?

Only in NLP there is some consensus for this: **https://gluebenchmark.com/**

What about in Vision?

Good metrics and representative tasks
are key to drive the field forward.

In NLP there has been some effort towards this: **http://www.statmt.org/**
http://www.parl.ai/

A. Wang et al. “GLUE: A multi-task benchmark and analysis platform for NLU” arXiv 1804:07461

Challenge #2: General Principle

Is there a **general** principle of unsupervised feature learning?

The current SoA in NLP: word2vec, BERT, etc. are **not entirely satisfactory** - very local predictions of a single missing token..

E.g.: This tutorial is … … because I learned … …!
Impute: This tutorial is **really awesome** because I learned **a lot**!
Feature extraction: topic={education, learning}, style={personal}, …

Ideally, we would like to be able to impute any missing information given some context, we would like to extract features describing any subset of input variables.

Challenge #2: General Principle

Is there a **general** principle of unsupervised feature learning?

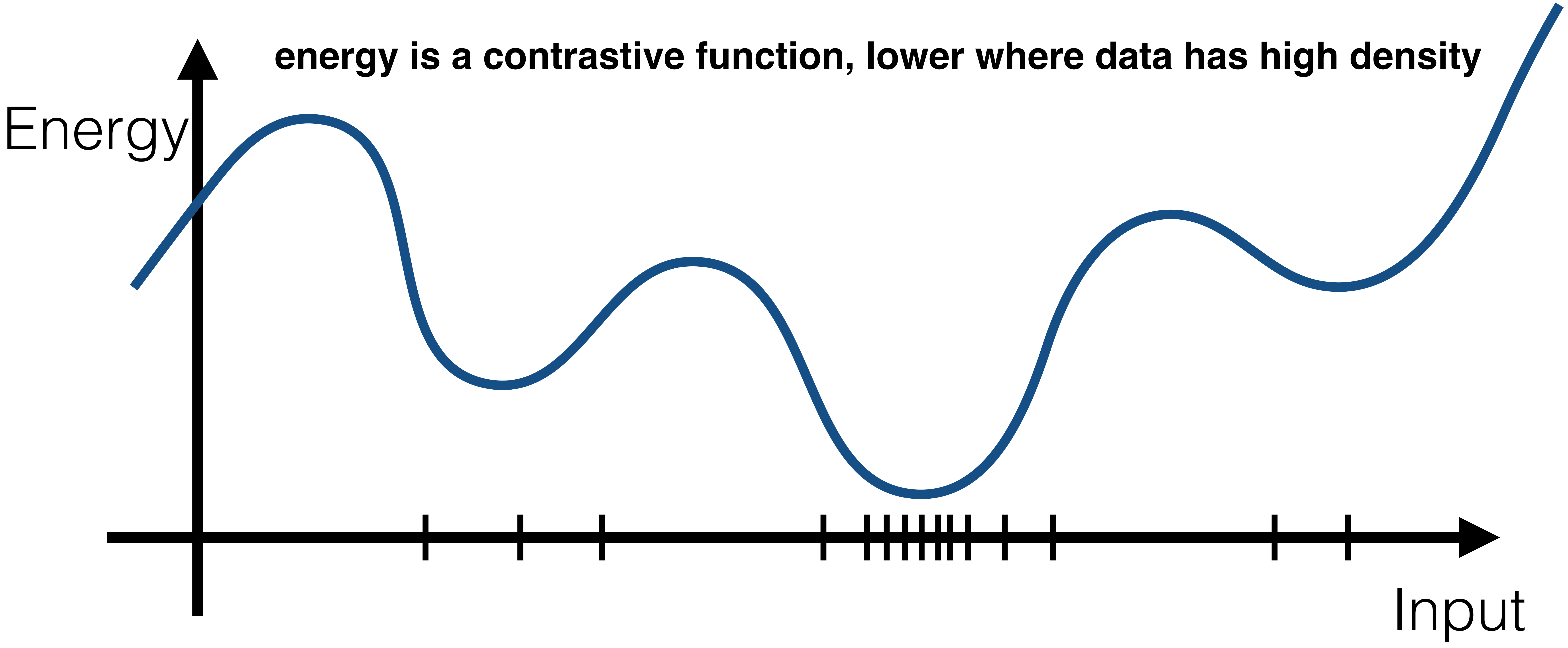
The current SoA in NLP: word2vec, BERT, etc. are **not entirely satisfactory** - very local predictions of a single missing token..

The current SoA in Vision: SSL is **not entirely satisfactory** - which auxiliary task and how many more tasks do we need to design?

Limitations of auto-regressive models: need to specify order among variables making some prediction tasks easier than others, slow at generation time.

Challenge #2: General Principle

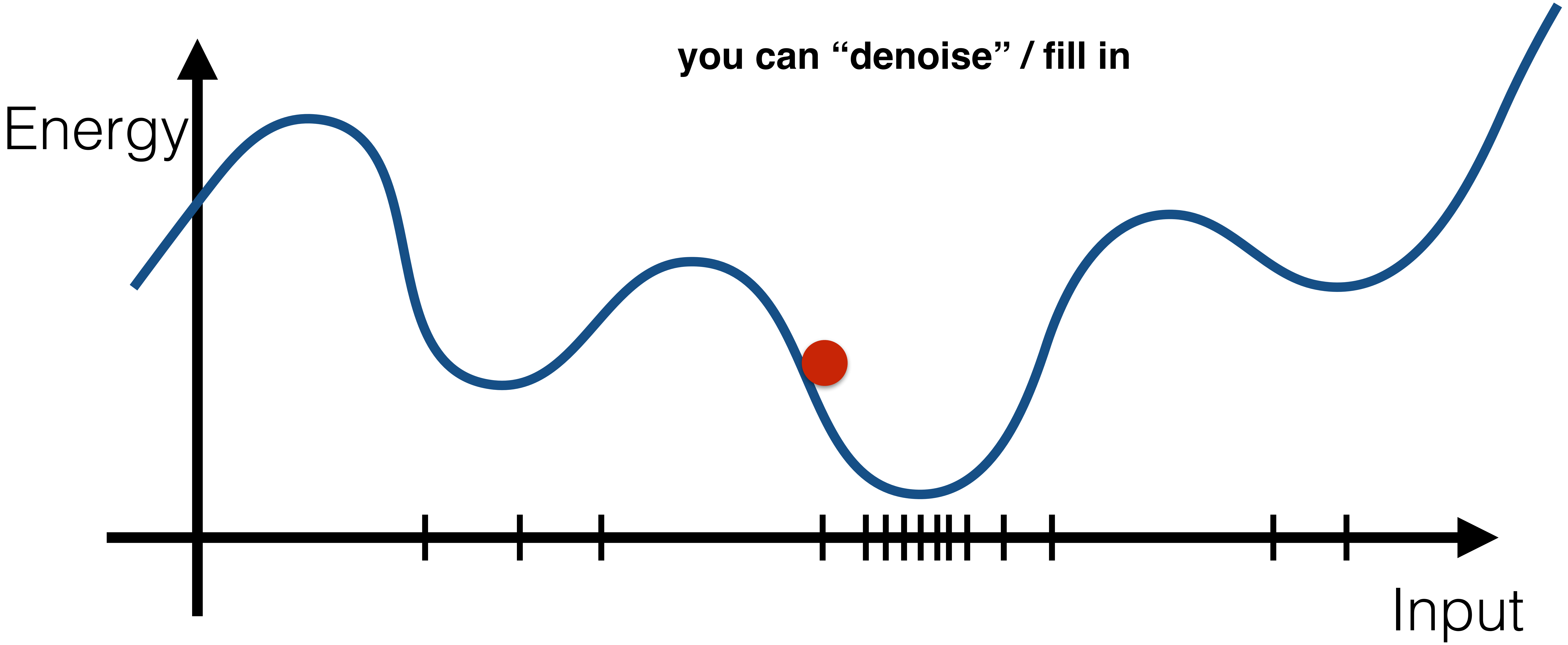
A brief case study of a more general framework: EBMs



Y. LeCun et al. “A tutorial on energy-based learning” MIT Press 2006

Challenge #2: General Principle

A brief case study of a more general framework: EBMs

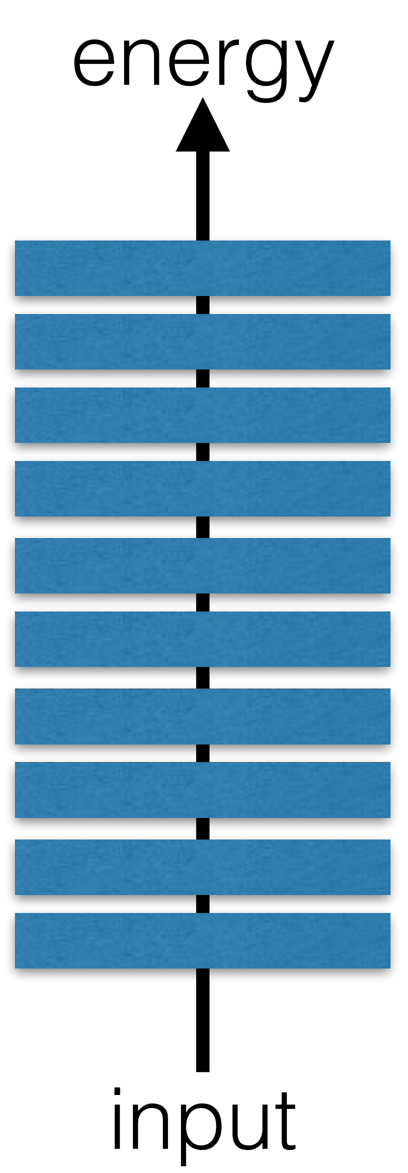


Y. LeCun et al. "A tutorial on energy-based learning" MIT Press 2006

Challenge #2: General Principle

One possibility: energy-based modeling

you can do feature extraction using any intermediate representation from E(x)



Y. LeCun et al. “A tutorial on energy-based learning” MIT Press 2006

Challenge #2: General Principle

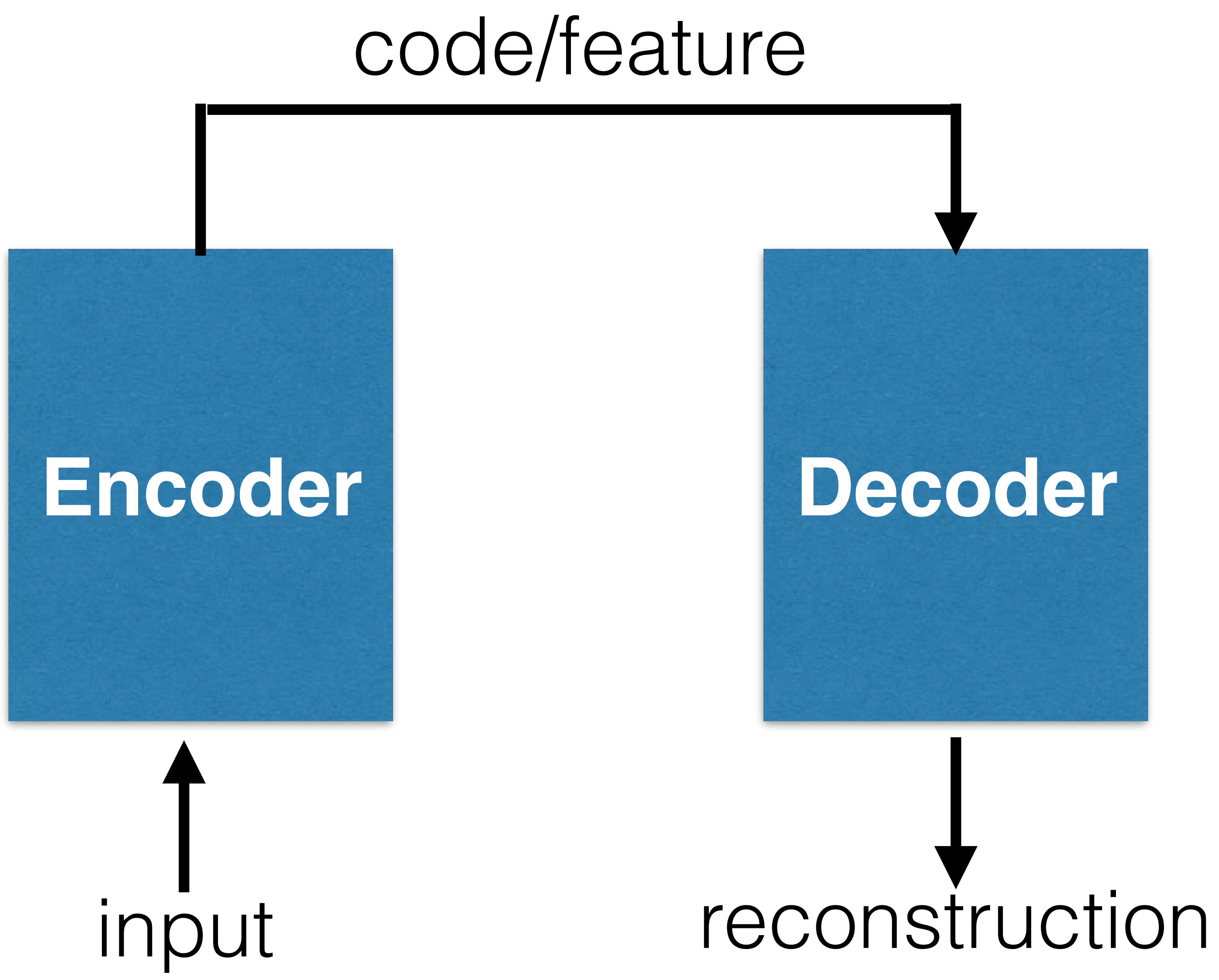
One possibility: energy-based modeling

The generality of the framework comes at a price…

Learning such contrastive function is in general very hard.

Y. LeCun et al. “A tutorial on energy-based learning” MIT Press 2006

Challenge #2: General Principle

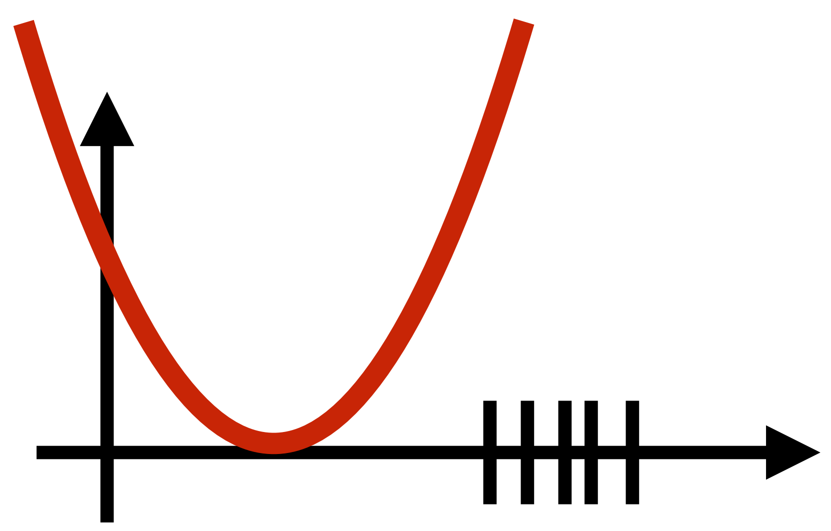


Learning contrastive energy function by pulling up on fantasized “negative data”:

- via search
- via sampling (*CD)

and/or by **limiting amount of information** going through the “code”:

- sparsity
- low-dimensionality
- noise



M. Ranzato et al. “A unified energy-based framework for unsupervised learning” AISTATS 2007
A. Hyvärinen “Estimation of non-normalized statistical models by score matching” JMNR 2005
K. Kavukcuoglu et al. “Fast inference in sparse coding algorithms…” arXiv 1406:5266 2008

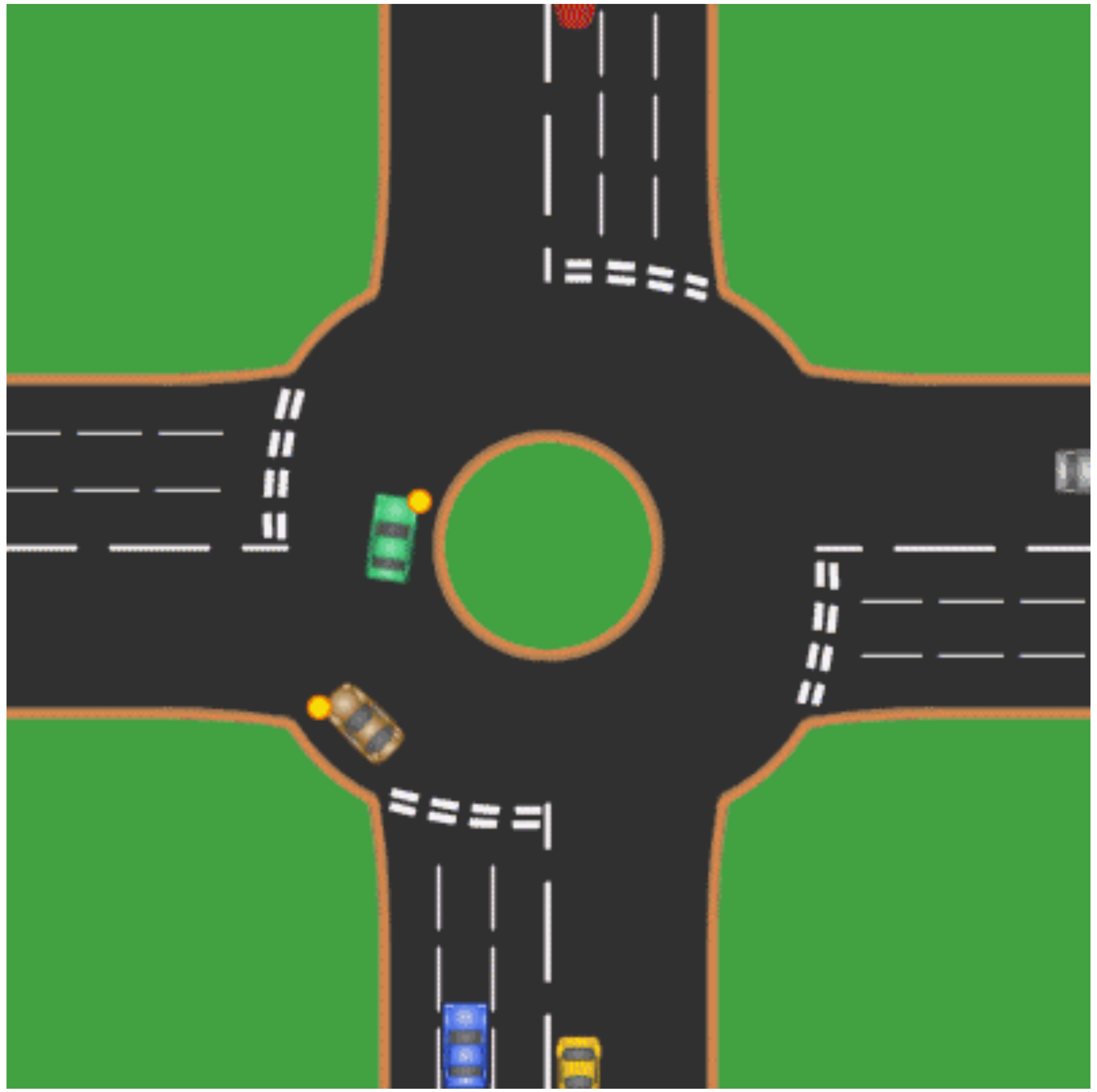
Challenge #2: General Principle

Challenge: If the space is very high-dimensional, it is difficult to figure out the right “pull-up” constraint that can properly shape the energy function.

- Are there better ways to pull up?
- Is there a better framework?
- To which extent should these principles be agnostic of the architecture and domain of interest?

Challenge #3: Modeling Uncertainty

- Most predictions tasks have uncertainty.



where is the red car going?

Challenge #3: Modeling Uncertainty

- Most predictions tasks have uncertainty.

E.g.: This tutorial is … … because I learned … …!

Impute: This tutorial is **really awesome** because I learned **a lot**!

This tutorial is **so bad** because I learned **really nothing**!

Challenge #3: Modeling Uncertainty

- Most predictions tasks have uncertainty.
- Several ways to model uncertainty:
    - latent variables
    - GANs
    - shaping energies to have lots of minima
    - quantizing continuous signals…

What are efficient ways to learn and do inference?

How to model uncertainty in continuous distributions?

The Big Picture

- A big challenge in AI: learning with less labeled data.
- Lots of sub-fields in ML tackling this problem from other angles:
    - few-shot learning
    - meta-learning
    - life-long learning
    - transfer learning
    - semisupervised
    - …
- Unsupervised learning is part of a broader effort.

The Big Picture

Unsupervised Learning should eventually be considered as a component within a bigger system.

- RL models can work more efficiently by leveraging information present in the input observations (unsupervised learning).
- Unsupervised learning is an important tool, but sparse rewards (RL) can inform about what unsupervised tasks are meaningful. Environment can provide further constraints.

you can't eat just the cherry, nor just the filling….
you gotta eat a whole slice!

picture/metaphor credit: Y. LeCun

Conclusions

- Unsupervised Learning is a key ingredient for any agent that learns from few interactions / few labeled examples.
- Lots of sub-areas: feature learning, learning to align domains, learning to generate samples, …
- Unsupervised learning currently works very well in restricted settings and in few applications.
- Biggest challenges:
    - metrics & tasks,
    - generality and efficiency of current algorithms,
    - integration of unsupervised learning with other learning components.

תודה
Dankie Gracias
Спасибо شكراً
Merci Takk
Köszönjük
Terima kasih
Grazie Dziękujemy Dėkojame
Ďakujeme Vielen Dank Paldies
Kiitos Täname teid 谢谢
Thank You
Tak
感謝您 Obrigado Teşekkür Ederiz
감사합니다
Σας Ευχαριστούμ
ขอบคุณ
Bedankt Děkujeme vám
ありがとうございます
Tack