Advanced Deep Learning: A Framework

Anders Søgaard

coAStaL

Course outline

Goal 1: Quick tour of recent developments in deep learning

Goal 2: Inspiration for thesis/research projects

Week	Lecturer	Subject	Literature	Assignment
1	Stefan	Introduction to Neural Networks.	d2l 2.1-2.5, 2.7, 11.5.1, **slides**	
2	Stefan	CNNs; FCNs; U-Nets. Data augmentation; invariance; regularization e.g. dropout	d2l 6, 7, 13.9-13.11, **slides**	Assignment 1 (May 10)
3	Anders/Phillip	May 9 (A): RNNs May 11 (P): Transformers	d2l 8 Transformers: d2l 10.5-10.7 + Vaswani et al. (2017)	Assignment 2 (May 20)
4	Phillip/Anders	May 16 (P): Representation and Adversarial Learning May 18 (A): A Learning Framework + Self-supervised Learning + Contrastive Learning	Autoencoders: blog post GANs: Goodfellow (2016) Self-supervised learning: blog post Contrastive learning: Dor et al. (2018) Adversarial examples: Goodfellow et al. (2015)	
5	Anders	May 23: General Properties, e.g., Scaling Laws, Lottery Tickets, Bottleneck Phenomena May 25: Applications of Representation, Adversarial and Contrastive Learning	GANs: Lample et al. (2018) Autoencoders: Chandar et al. (2011) Contrastive learning: Yu et al. (2018) DynaBench: Talk by Douwe Kiela (Facebook, now HuggingFace) Scaling laws: Kaplan et al. (2020)	Assignment 3 *[MC on Representation Learning/1p Report on Lottery Ticket extraction]* (June 3)
6	Anders	May 30: Interpretability, Transparency, and Trustworthiness & Deep Learning for Scientific Discovery June 1: Interpretability (Feature Attribution), including Guest Lecture by Stephanie Brandl	DL for Scientific Discovery: Sullivan (2022) Interpretability/Background: Søgaard (2022)	
7	Anders	June 6: *Off (no teaching)* June 8: Interpretability (Training Data Influence)	Literature: Feng and Boyd-Graber (2018); Jiang and Senge (2021)	
8	Anders	June 13-15: Best Practices	Literature: Dodge et al. (2019) and Raji et al. (2021)	Assignment 4 *[MC on Interpretability; 1p Report on Best Practices]* (June 21)

Today

a) (Traditional ML) Landscape
b) (Non-Contrastive) Self-Supervised Learning
c) Contrastive Learning
d) Connections to Semi-Supervised Learning and Domain Adaptation
e) Landscape (revisited)
f) Pretraining in Practice

Landscape

Standard ML Framework

1. Supervised Learning (Estimating functions from X to Y)
    a. Classification
    b. Regression
2. Unsupervised Learning (Learning similarity to group X)
3. Semi-supervised Learning

Other dimension: Generative vs. Discriminative

Standard ML Framework

1. Supervised Learning (Estimating functions from X to Y)
    a. Classification
    b. Regression
2. Unsupervised Learning (Learning similarity to group X)
3. Semi-supervised Learning

Other dimension: Generative vs. Discriminative

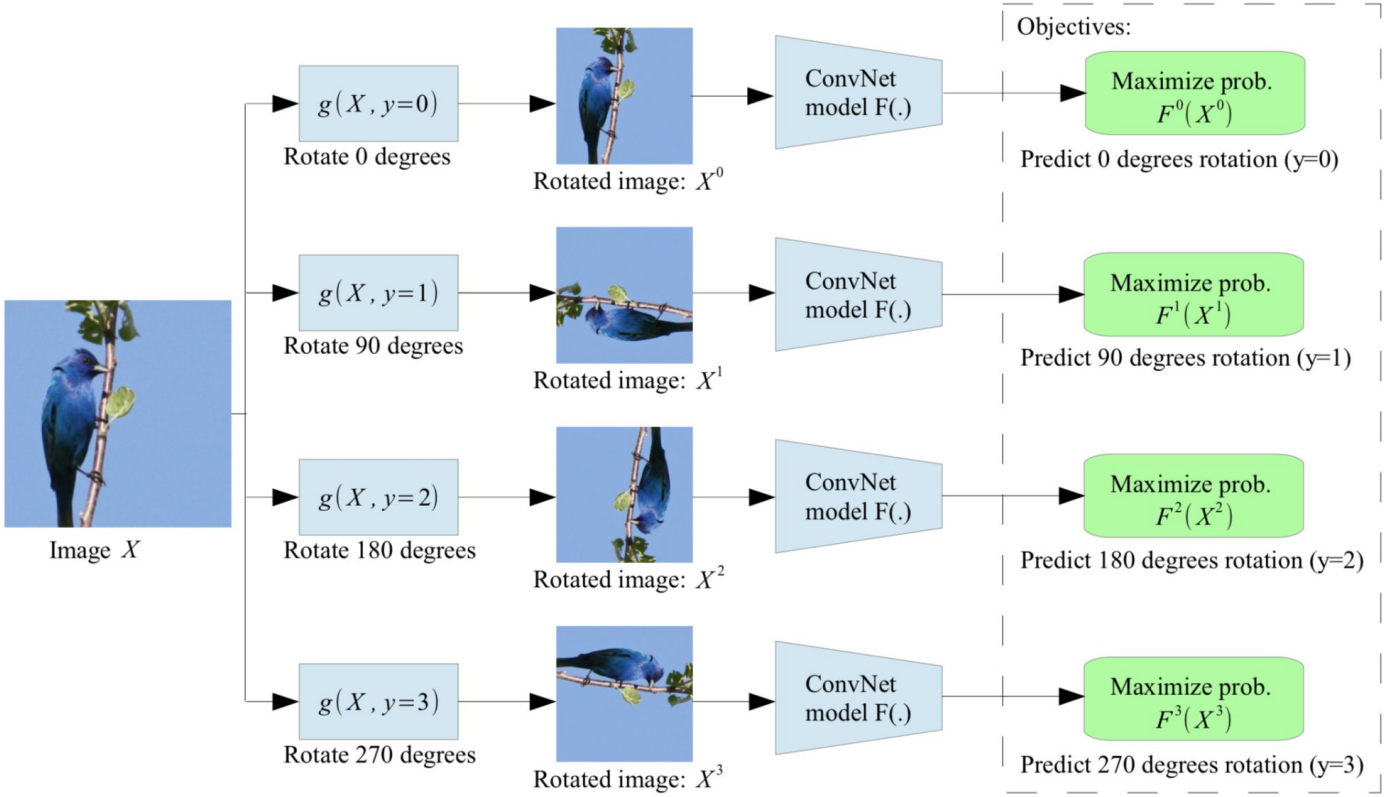
What about Multi-Task Learning?

Standard ML Framework

1. Supervised Learning (Estimating functions from X to Y)
    a. Classification
    b. Regression
2. Unsupervised Learning (Learning similarity to group X)
3. Semi-supervised Learning

Other dimension: Generative vs. Discriminative

What about Self-Supervised and Contrastive Learning?



Example of self-supervised learning

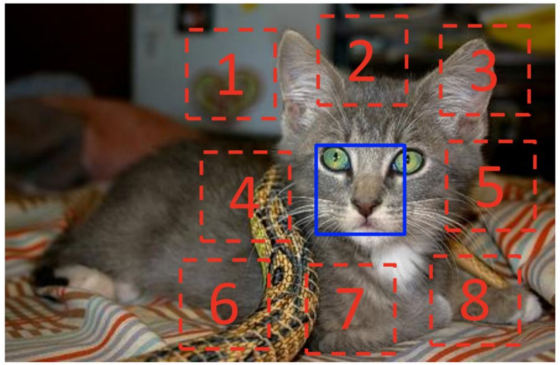
(Non-Contrastive) Self-supervised Learning

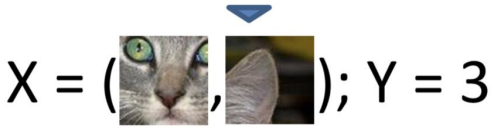
Self-Supervised Learning

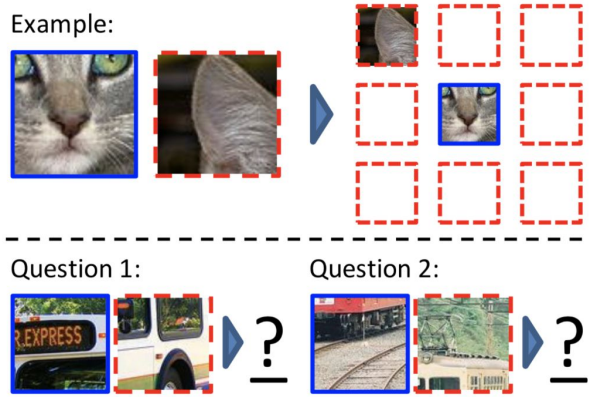
Contrastive	Non-contrastive
Input is pairs or triples of samples	Input is single samples
Uses positive and negative samples	Uses only positive samples
Relies on a method for sampling negatives	Relies on a perturbation function

Predict position

Strategy: Predict the position of different patches of the image.



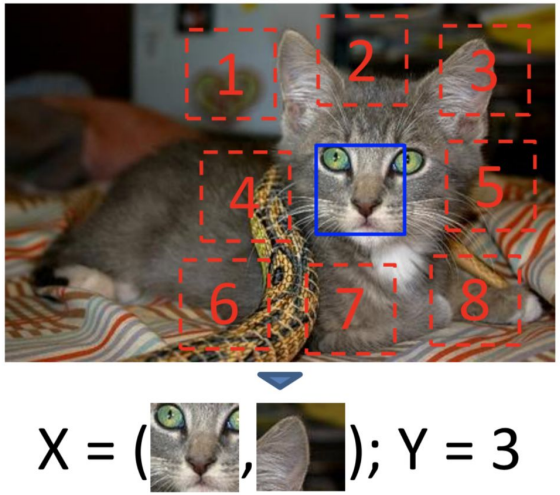


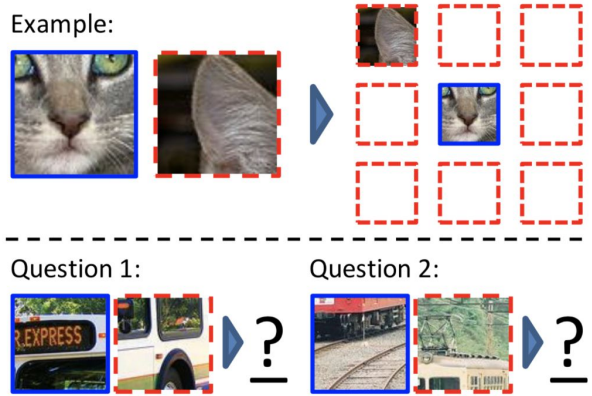


Predict position

Strategy: Predict the position of different patches of the image.

Variation: Predict the spatial relation between two or more patches.



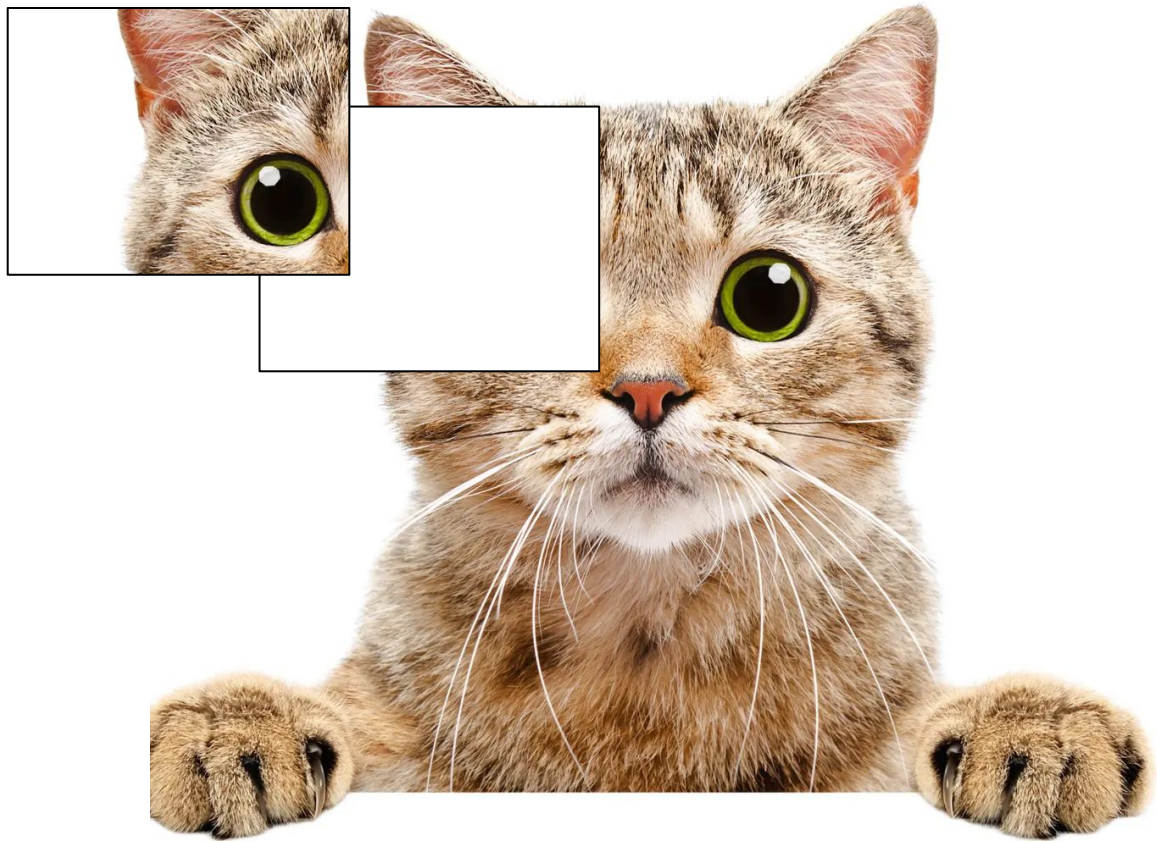


Predict part

Strategy: Predict a patch from the rest of the image.

Predict part

Strategy: Predict a patch from the rest of the image.



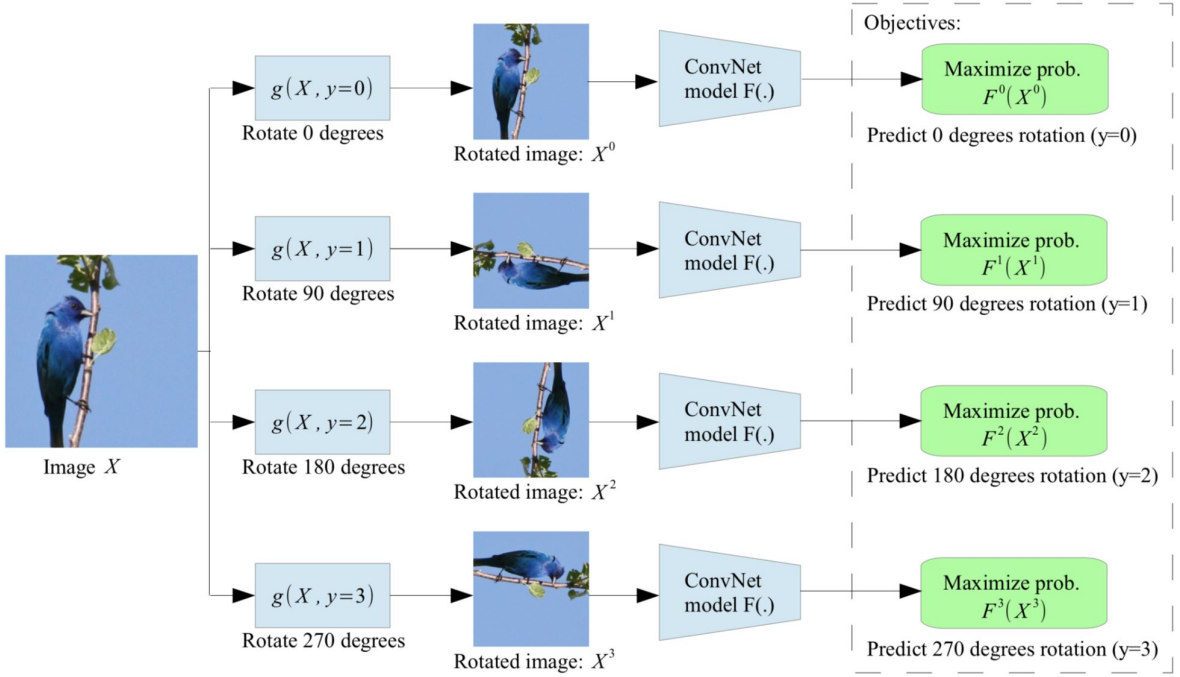
Predict corruption

Strategy: Predict a patch that is corrupted.

Exercise: Where is the corruption?

Predict rotation

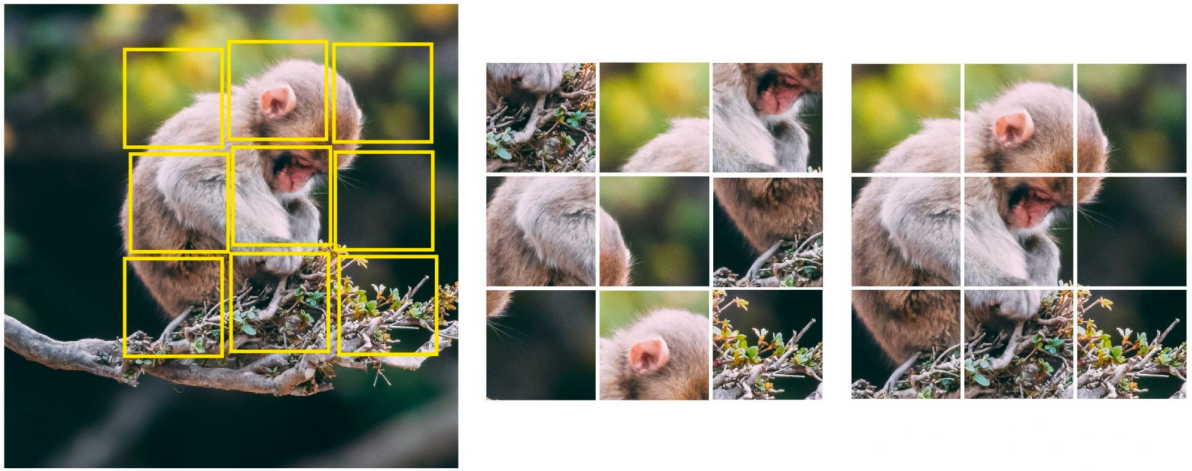
Predict the natural orientation of an image.



Predict reordering

Strategy: Predict the proper reordering of permuted patches.

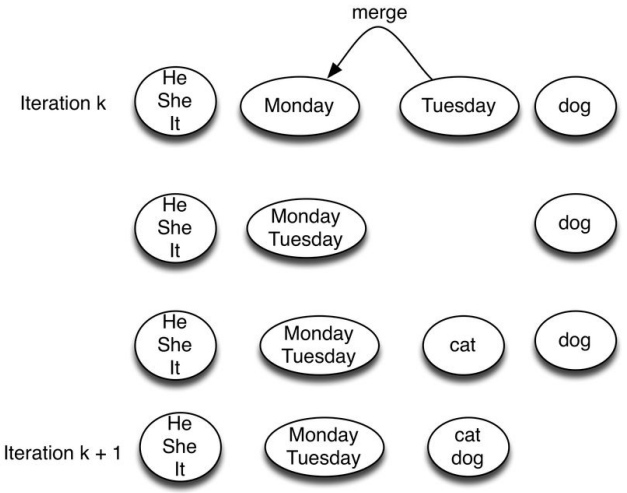
Also known as the jigsaw puzzle pretraining strategy.

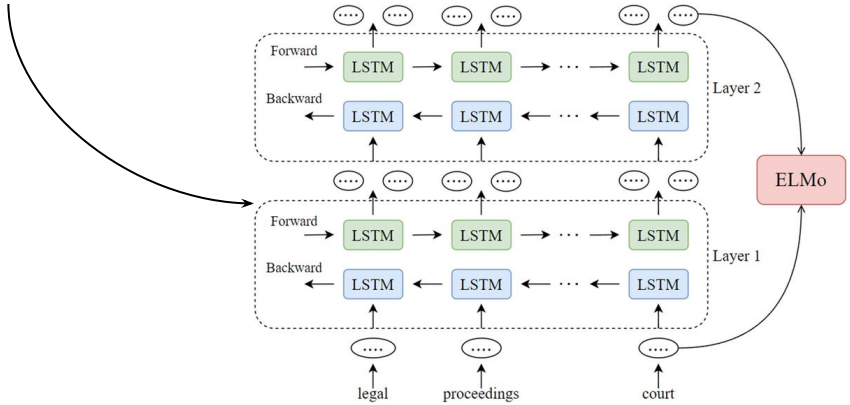


Most Non-Contrastive Self-Supervised NLP is ‘predict part’

From Brown to ELMO

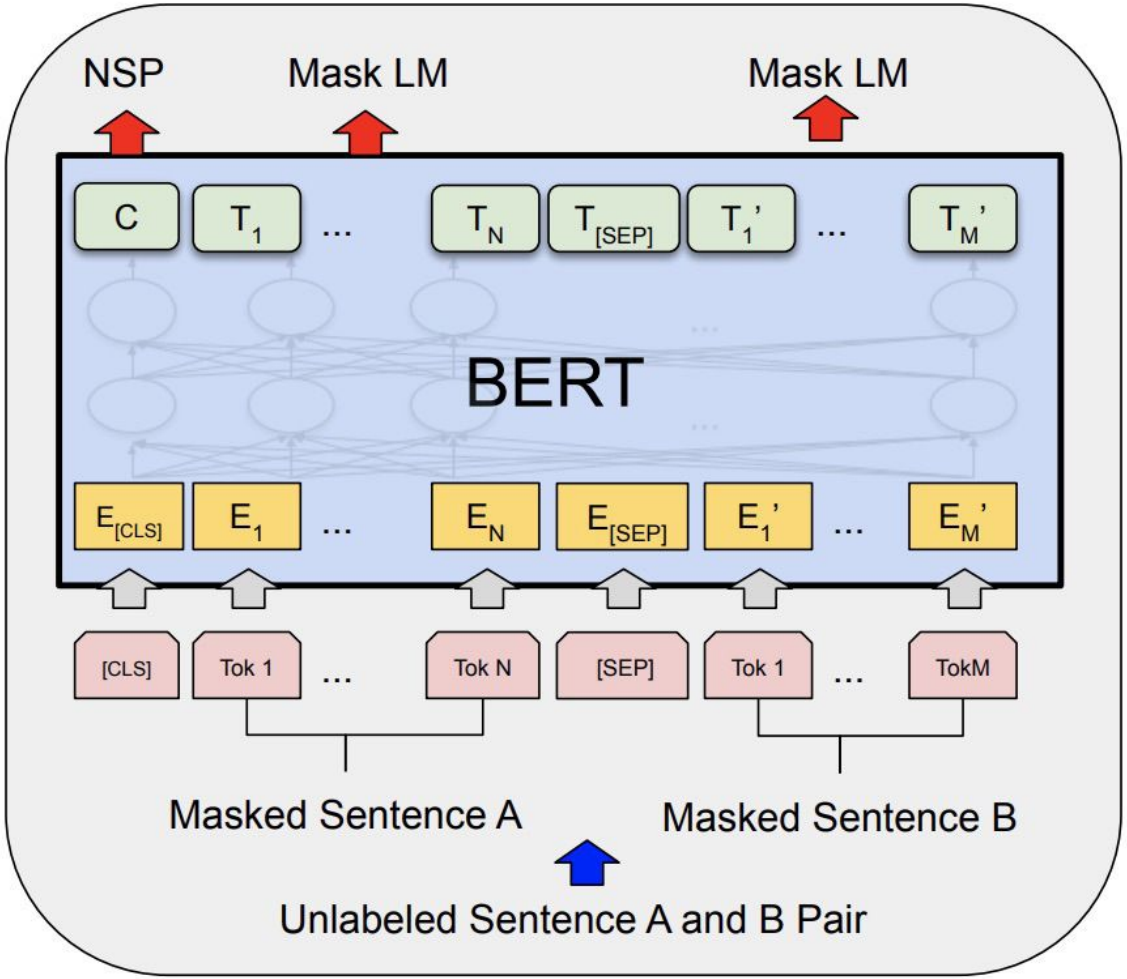
Both are so-called **auto-regressive** language models (bigram and LSTM, respectively). Brown outputs *discrete* clusters, whereas ELMO outputs *continuous* vectors.





BERT

BERT is **non-autoregressive**, predicting random (masked) words in context, and relying on the parallel processing of multi-headed attention blocks.



Variations of BERT

Since 2017, NLP has been all about the BERT, and 100s of variations over BERT has seen the day.

RoBERTa	Bigger, different hyper-parameters, no NSP
ELECTRA	Uses simply model to prune candidates for masked tokens
mBERT	Trained on the concatenation of corpora in different languages
ERNIE	Trained on text and knowledge graphs
HierBERT	Hierarchical architecture, i.e., BERT over BERTs
TOD-BERT, EstBERT, FlauBERT, RoBERT, tinyBERT, LIMIT-BERT, E-BERT, LadaBERT, spanBERT, schuBERT, BERT-kNN, etc.	Different applications, different domains, different languages, different ML tricks, more efficient versions, etc.

Exercise

What would Non-Contrastive Self-Supervised NLP that **was not** ‘predict part’ look like?

Exercise

What would Non-Contrastive Self-Supervised NLP that **was not** 'predict part' look like?

Solutions proposed elsewhere: local reordering, global reordering, shuffled token detection. The first two are jigsaw puzzle pretraining strategies, the third corruption. *Note:* No obvious equivalent to 'predict rotation' in language, and 'predict position' probably too easy.

An Empirical Exploration of Local Ordering Pre-training for Structured Prediction

Zhisong Zhang, Xiang Kong, Lori Levin, Eduard Hovy
Language Technologies Institute, Carnegie Mellon University
{zhisongz,xiangk,lsl,hovy}@cs.cmu.edu

SLM: Learning a Discourse Language Representation with Sentence Unshuffling

Haejun Lee♡ **Drew A. Hudson**♣ **Kangwook Lee**♡ **Christopher D. Manning**♣
♡ Samsung Research ♣ Stanford University
{haejun82.lee, kw.brian.lee}@samsung.com
{dorarad, manning}@cs.stanford.edu

Shuffled-token Detection for Refining Pre-trained RoBERTa

Subhadarshi Panda
Graduate Center
CUNY
spanda@gc.cuny.edu

Anjali Agrawal
New York University
aa7513@nyu.edu

Jeewon Ha
New York University
jh6926@nyu.edu

Benjamin Bloch
New York University
bb1976@nyu.edu

Exercise

What would Non-Contrastive Self-Supervised NLP that **was not** 'predict part' look like?

Solutions proposed elsewhere: local reordering, global reordering, shuffled token detection. The first two are jigsaw puzzle pretraining strategies, the third corruption. *Note:* No obvious equivalent to 'predict rotation' in language, and 'predict position' probably too easy.

ALBERT uses sentence reordering objective!

An Empirical Exploration of Local Ordering Pre-training for Structured Prediction

Zhisong Zhang, Xiang Kong, Lori Levin, Eduard Hovy
Language Technologies Institute, Carnegie Mellon University
{zhisongz,xiangk,lsl,hovy}@cs.cmu.edu

SLM: Learning a Discourse Language Representation with Sentence Unshuffling

Haejun Lee♡ **Drew A. Hudson**♣ **Kangwook Lee**♡ **Christopher D. Manning**♣
♡ Samsung Research ♣ Stanford University
{haejun82.lee, kw.brian.lee}@samsung.com
{dorarad, manning}@cs.stanford.edu

Shuffled-token Detection for Refining Pre-trained RoBERTa

Subhadarshi Panda
Graduate Center
CUNY
spanda@gc.cuny.edu

Anjali Agrawal
New York University
aa7513@nyu.edu

Jeewon Ha
New York University
jh6926@nyu.edu

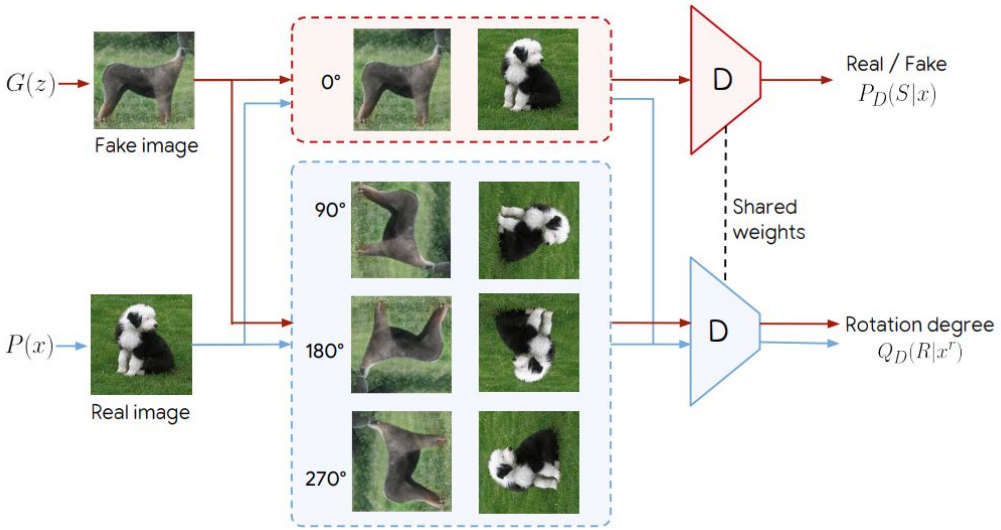
Benjamin Bloch
New York University
bb1976@nyu.edu

Contrastive Learning

How to get your fakes (negatives)

You can apply roughly the same perturbations we used to generate the input of non-contrastive self-supervised pretraining data, to generate an image pair for contrastive learning. You can also, however, use:

- Synthetic data
- Other data sources (that you do not care about)
- Other domains, languages,
- Less learnable transformations



Negatives from noise

Core idea: Use a simple noise injection strategy, e.g., drop-out, and decide which sample contains noise.

Note: Similar to denoising autoencoders.

Example: Sim-CSE

Positive	**Negative**
I like Ghana drill	I Ghana drill

Negatives from substitution (NLP)

Core idea: Use lexical substitution to obtain negatives

Note: Similar to ELECTRA pretraining.

Example: This paper

Positive	Negative
I like Ghana drill	I like Rwanda drill

Auto-generated negatives

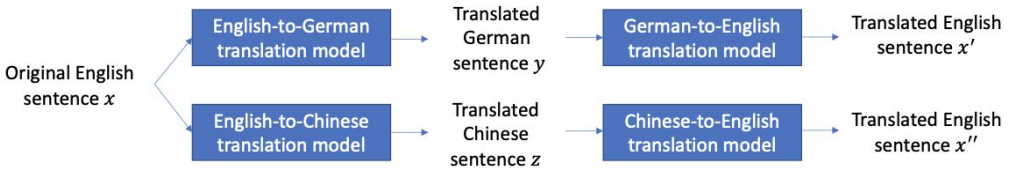
Core idea: Use an imperfect model to generate data and use metric to filter away the best data. Such generated data can provide useful negative data, e.g., for what an image or text does not look like.

Example: CLIFF, GPT-2, using injected grammatical mistakes, etc.

Auto-generated negatives

Core idea: Use an imperfect model to generate data and use metric to filter away the best data. Such generated data can provide useful negative data, e.g., for what an image or text does not look like.

Example: CLIFF, GPT-2, using injected grammatical mistakes, etc.



Creative approach from UCSD researchers. Using round-trip translation to generative negatives.

Exercise: What's the limitation of this approach?

Exercise

What other methods for findings negatives can you think of?

We thought of adding noise, using lexical substitutes (NLP), using imperfect data generators. What else is there?

Connections to Semi-Supervised Learning and Domain Adaptation

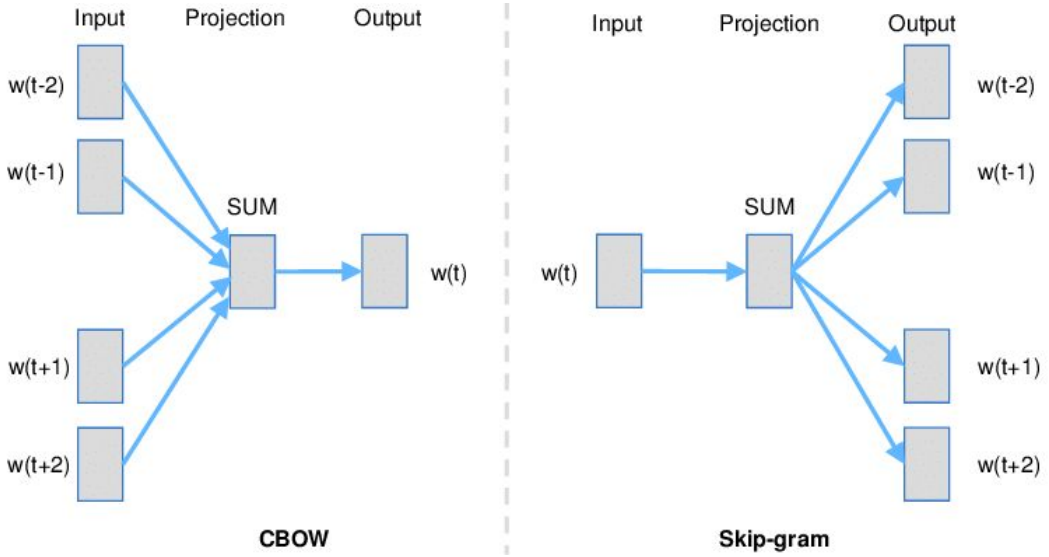
Quick brush-off

First-unlabeled-then-labeled

1. Clusters-as-features
2. Word embeddings

First-labeled-then-unlabeled

3. Self-training
4. Co-training
5. Tri-training
6. Expectation Maximization
7. Co-EM



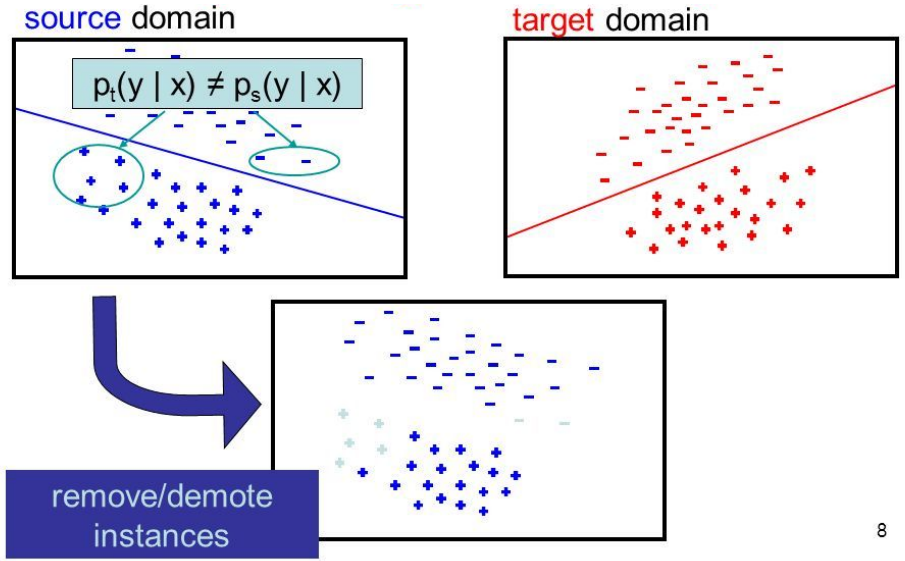
Note: CBOW and Skip-gram are linear predict-part methods.

Domain adaptation

Unsupervised domain adaptation - in which you have **labeled out-of-domain** and **unlabeled in-domain** data - comes in three flavors:

1. Semi-supervised learning
2. Instance weighting
3. Representation projection

3) E.g., GANs, Procrustes (next time).



Note: Instance weighting is a contrastive methods.

Landscape (revisited)

Standard ML Framework

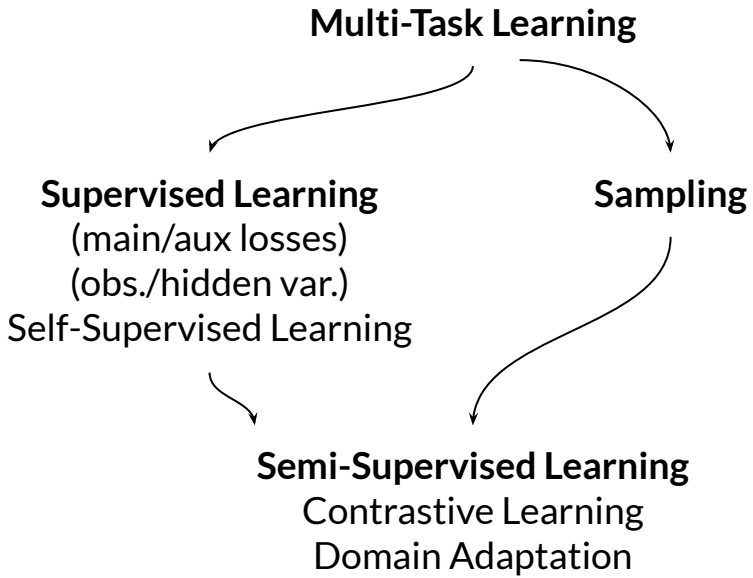
1. Supervised Learning (Estimating functions from X to Y)
    a. Classification
    b. Regression
2. Unsupervised Learning (Learning similarity to group X)
3. Semi-supervised Learning

Other dimension: Generative vs. Discriminative

MTL Framework

Observations:

- Supervised to Self-Supervised forms a continuum
- Contrastive Learning and (Unsupervised) Domain Adaptation are Semi-Supervised
- Multi-Player Architectures are easily formalized as MTL



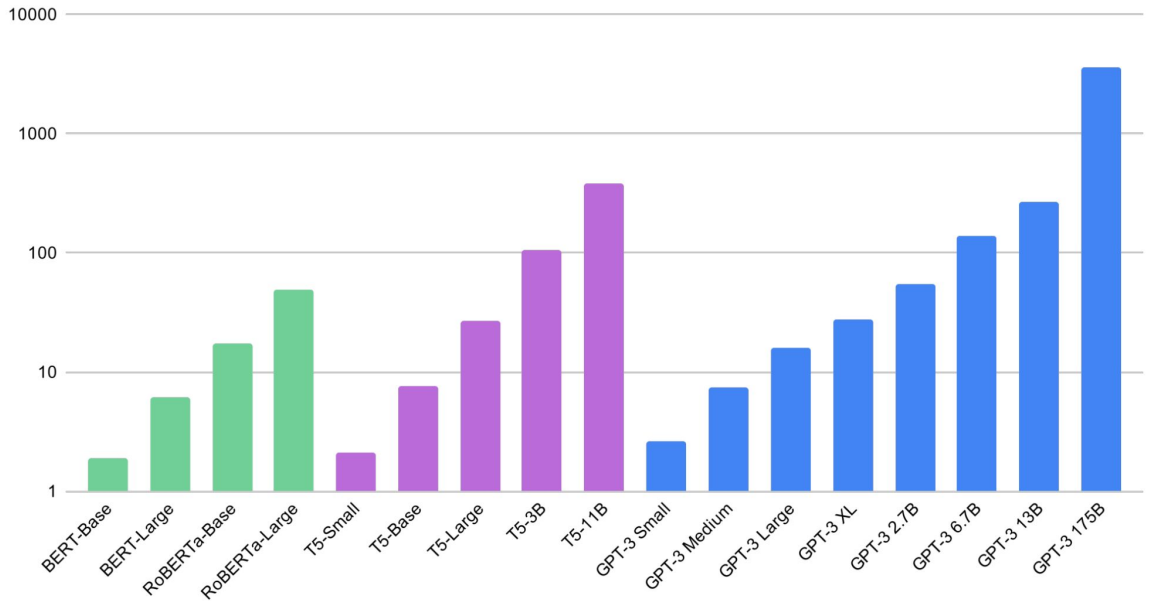
Pretraining in Practice

Model	Layers	Parameters	Hidden layer size	Training data	Objective
BERT-base	12	108m	768	16GB	MLM+NSP
BERT-large	24	324m	1024	16GB	MLM+NSP
ALBERT-base	12	12m	768	16GB	MLM+SRO
ALBERT-large	24	18m	1024	16GB	MLM+SRO
RoBERTa-large	24	324m	1024	160GB	MLM
GPT2	48	1542m	1600	40GB	Autoregressive
GPT3	96	170b	12288	570GB	Autoregressive

Other differences, e.g.: RoBERTa used a batch size of 8,000 with 300,000 steps. In comparison, BERT uses a batch size of 256 with 1 million steps

Compute hours

Measured in so-called ‘petaflop-days’ (~10^{20} operations)



Compute hours

Measured in so-called ‘petaflop-days’ (~10^{20} operations)