Machine Learning Beyond Static Datasets

ESSAI 2023

Dr. Martin Mundt,

Research Group Leader, TU Darmstadt & hessian.AI

Board Member of Directors, ContinualAI

Course: http://owll-lab.com/teaching/essai-23

hessian.AI

TECHNISCHE UNIVERSITÄT DARMSTADT

safe Financial Big Data Cluster

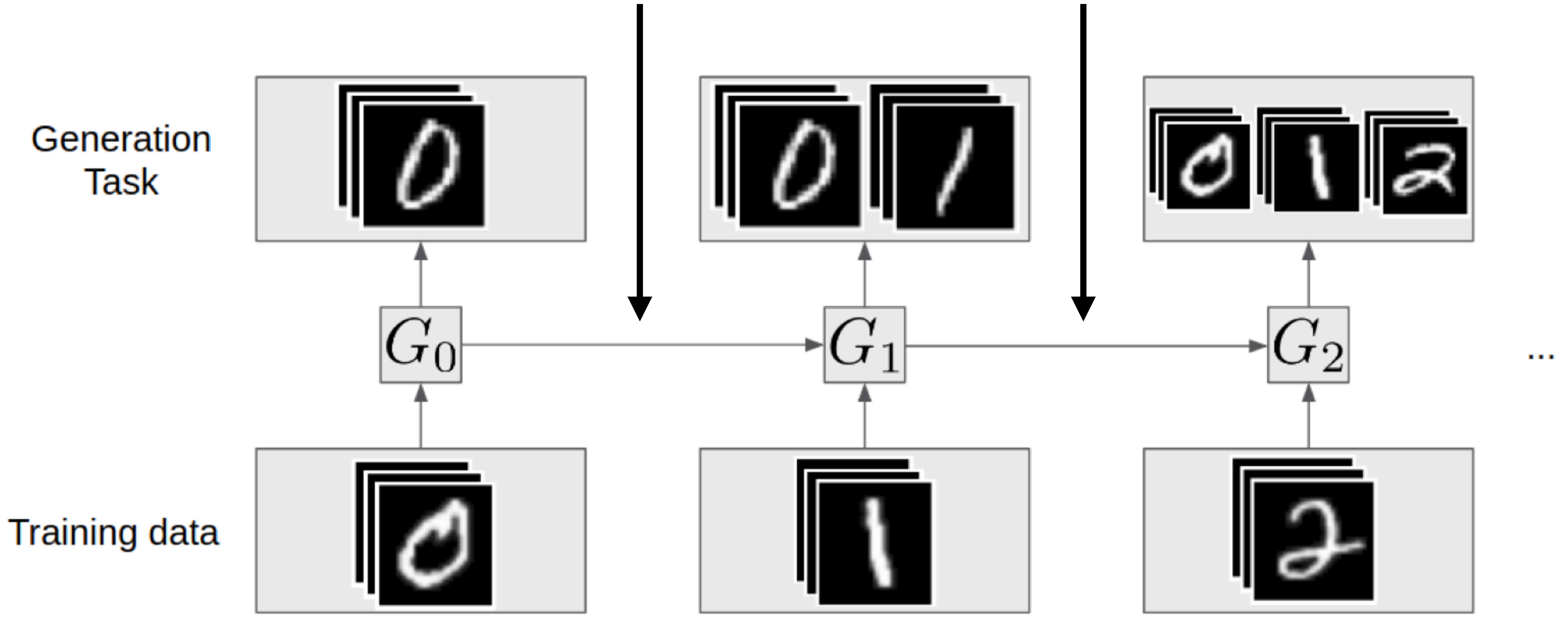
ContinualAI

QUEER in AI

Day 4: The Future
Data Selection & Learning Curricula

The future: picking what comes next

Who decides what comes next? A stream? A human? The model? How?



Lesort et al, "Generative Models from the perspective of Continual Learning", IJCNN 2019

Active learning

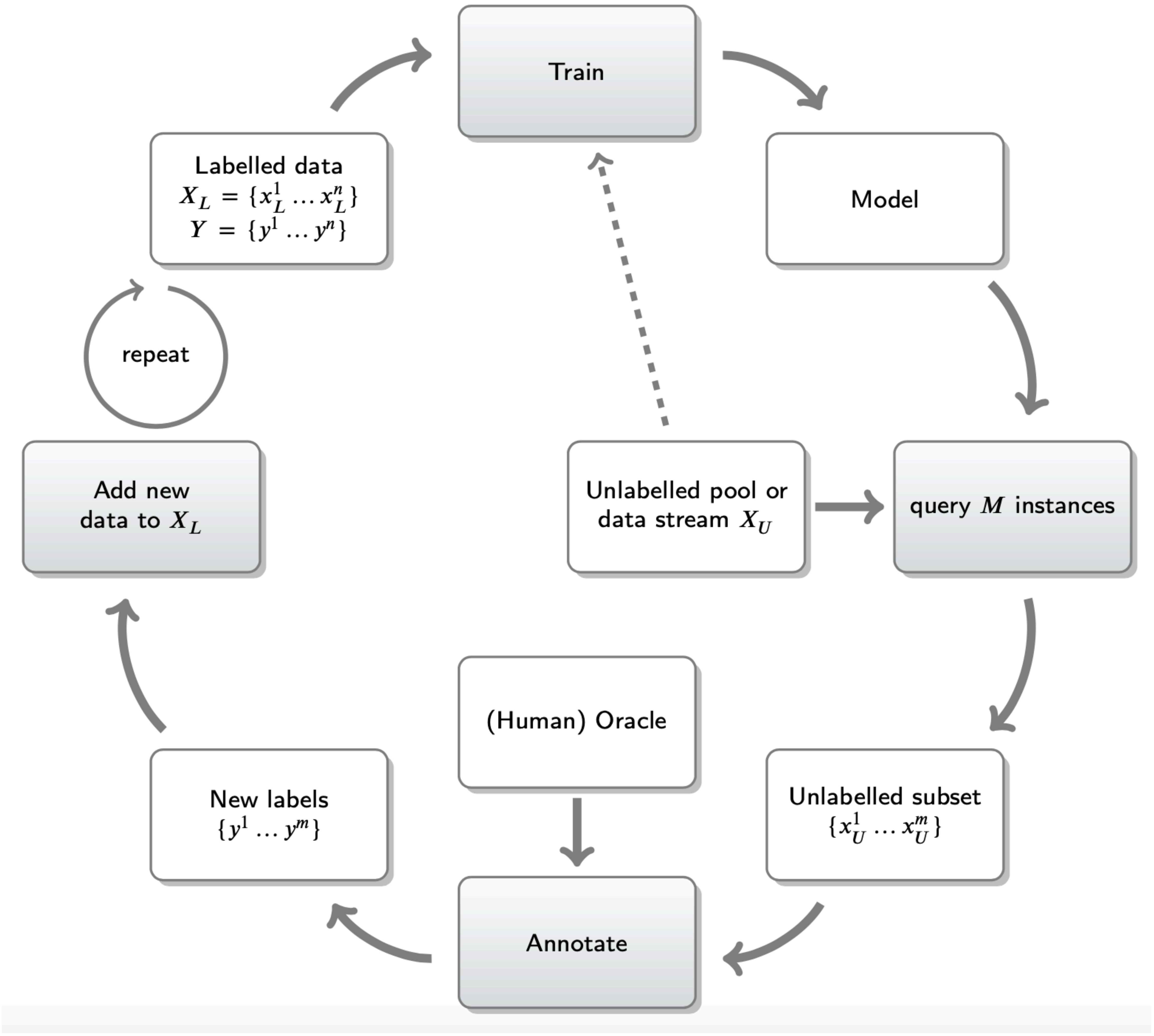


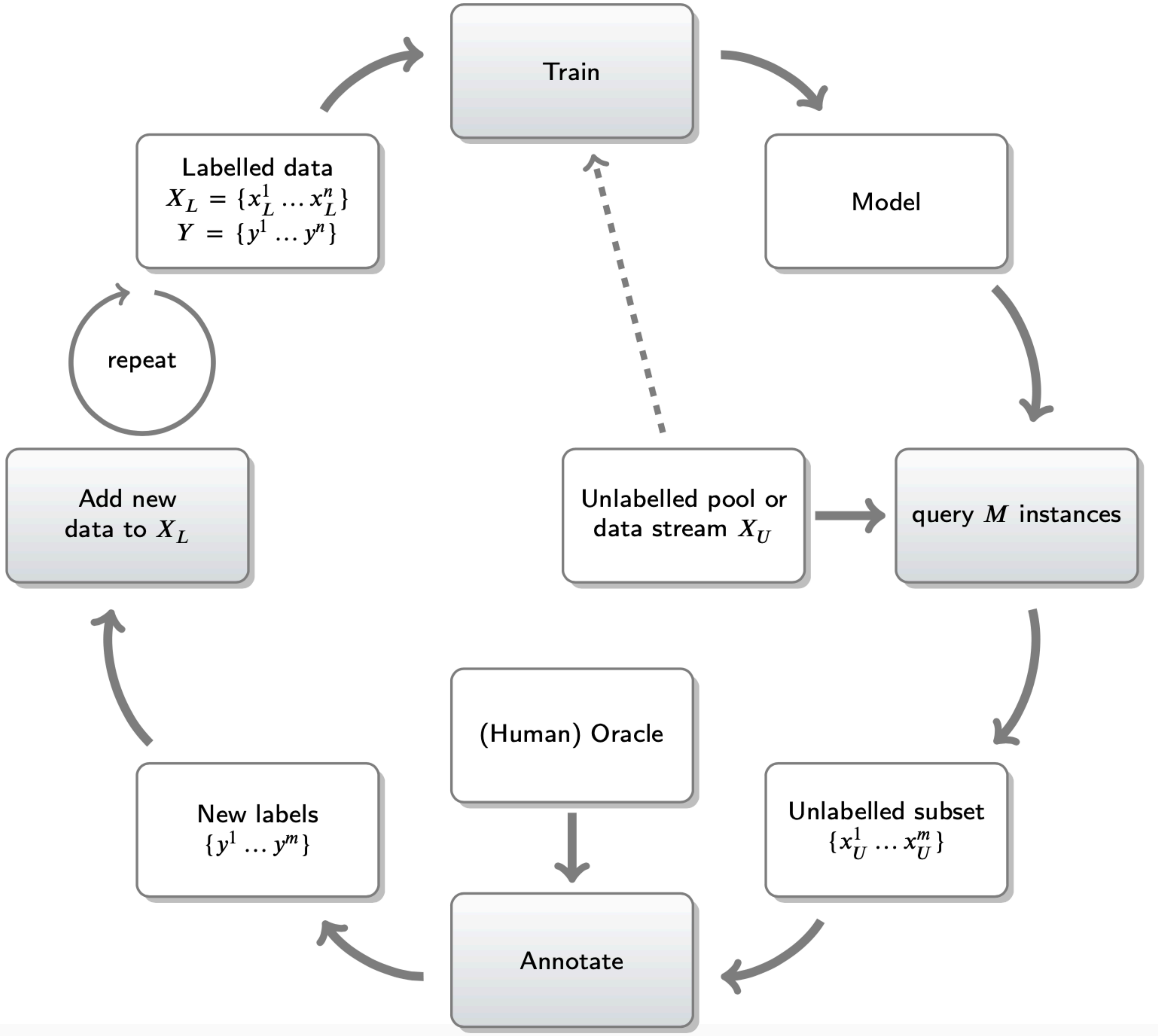
Figure from Mundt et al, “A Wholistic View of Continual Learning with Deep Neural Networks: Forgotten Lessons and the Bridge to Active and Open World Learning, Neural Networks 160, 2023

In essence: deciding what new data points are most informative

Also called “query learning” with the underlying mechanism called “acquisition function”

When querying new data, what are some assumptions & considerations on set-up we can make?

Active learning

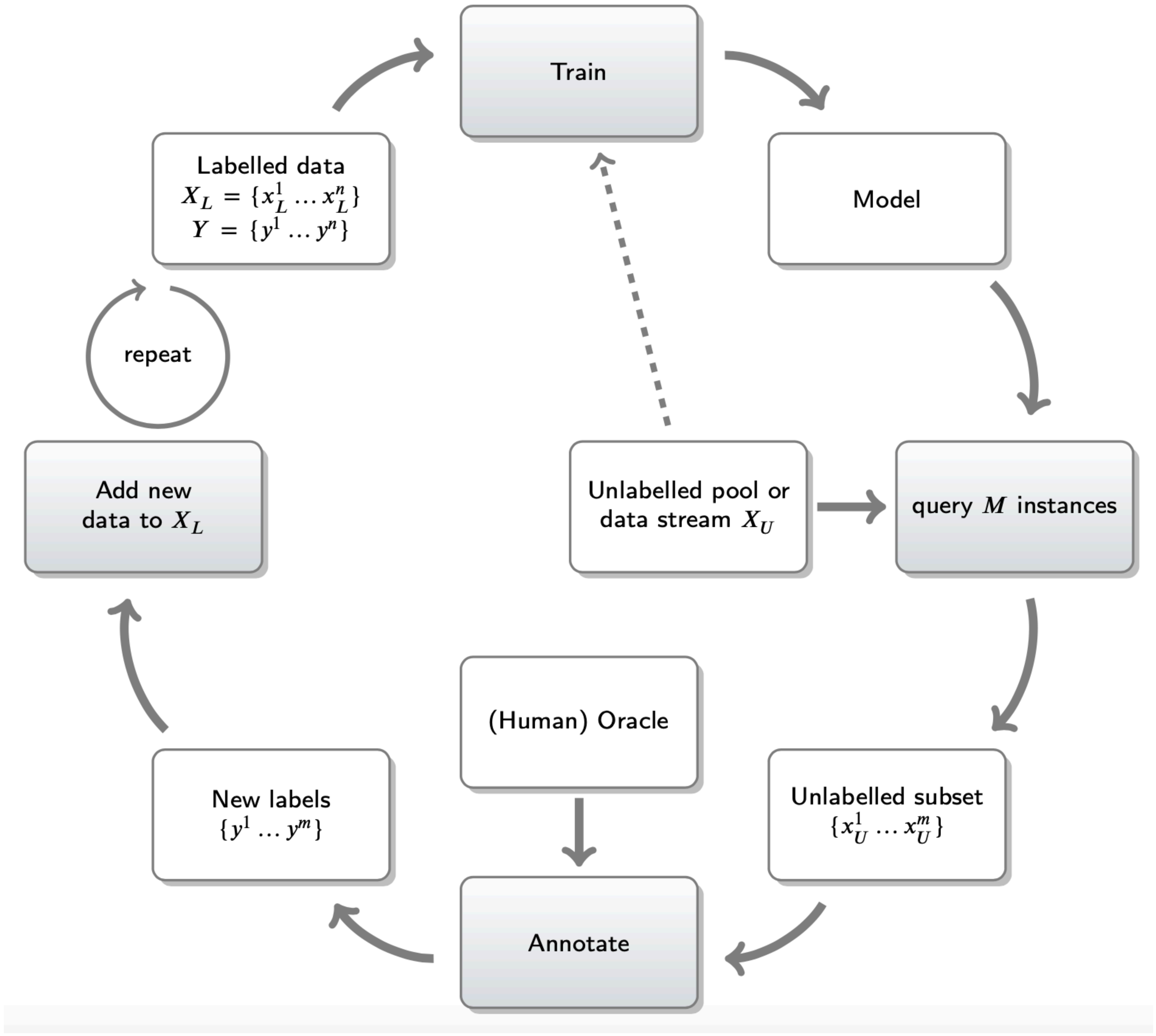


Many assumptions (non-exhaustive)

- Data is cheap vs. labeling is not?
- Pool of data upfront vs. stream?
- 1 data point vs. batches vs. tasks?
- Accumulate data after selection?
- Re-train vs. continued training?
- Oracle: infallible vs. noisy?

Figure from Mundt et al, "A Wholistic View of Continual Learning with Deep Neural Networks: Forgotten Lessons and the Bridge to Active and Open World Learning, Neural Networks 160, 2023

Majority of “traditional" active learning



Many assumptions (non-exhaustive)

- Data is cheap vs. labeling is not?
- Pool of data upfront vs. stream?
- 1 data point vs. batches vs. tasks?
- Accumulate data after selection?
- Re-train vs. continued training?
- Oracle: infallible vs. noisy?

Figure from Mundt et al, “A Wholistic View of Continual Learning with Deep Neural Networks: Forgotten Lessons and the Bridge to Active and Open World Learning, Neural Networks 160, 2023

What techniques to query data can you think of?

Again, a small tangent: discriminative or generative models?

Discriminative models could allow for natural ways to assess "novelty" of a new example -> *But caution*: overconfidence phenomena (tomorrow)

Generative models learn about the data distribution

-> *But caution*: our parameters only reflect the distribution seen so far!

(do we make use of a pool that is always available?)

We will see that the choice also depends on the set-up assumption!

See Zhang & Oles, "A Probability Analysis on the value of Unlabeled Data for Classification Problems", ICML 2000

Active learning perspectives

Version space reduction

reduce the set/space of possible hypotheses $h : \mathscr{X} \to \mathscr{Y}$ by removing the ones that are inconsistent with the data

Uncertainty & heuristics

use the predictions, or maybe even better, uncertainty in the predictions

Version Space

Version space (Mitchel 1978)

- Assume that there exist hypotheses consistent with the labeled data points $h : \mathcal{X} \to \mathcal{Y}$

  version space: $VS(D) = \{h \in H \,|\, \texttt{cons}(h, D)\}$

Figure from https://en.wikipedia.org/wiki/File:Version_space.png in the public domain

Version space (Mitchel 1978)

- Assume that there exist hypotheses consistent with the labeled data points $h : \mathscr{X} \to \mathscr{Y}$

  version space: $VS(D) = \{h \in H \,|\, \mathtt{cons}(h, D)\}$

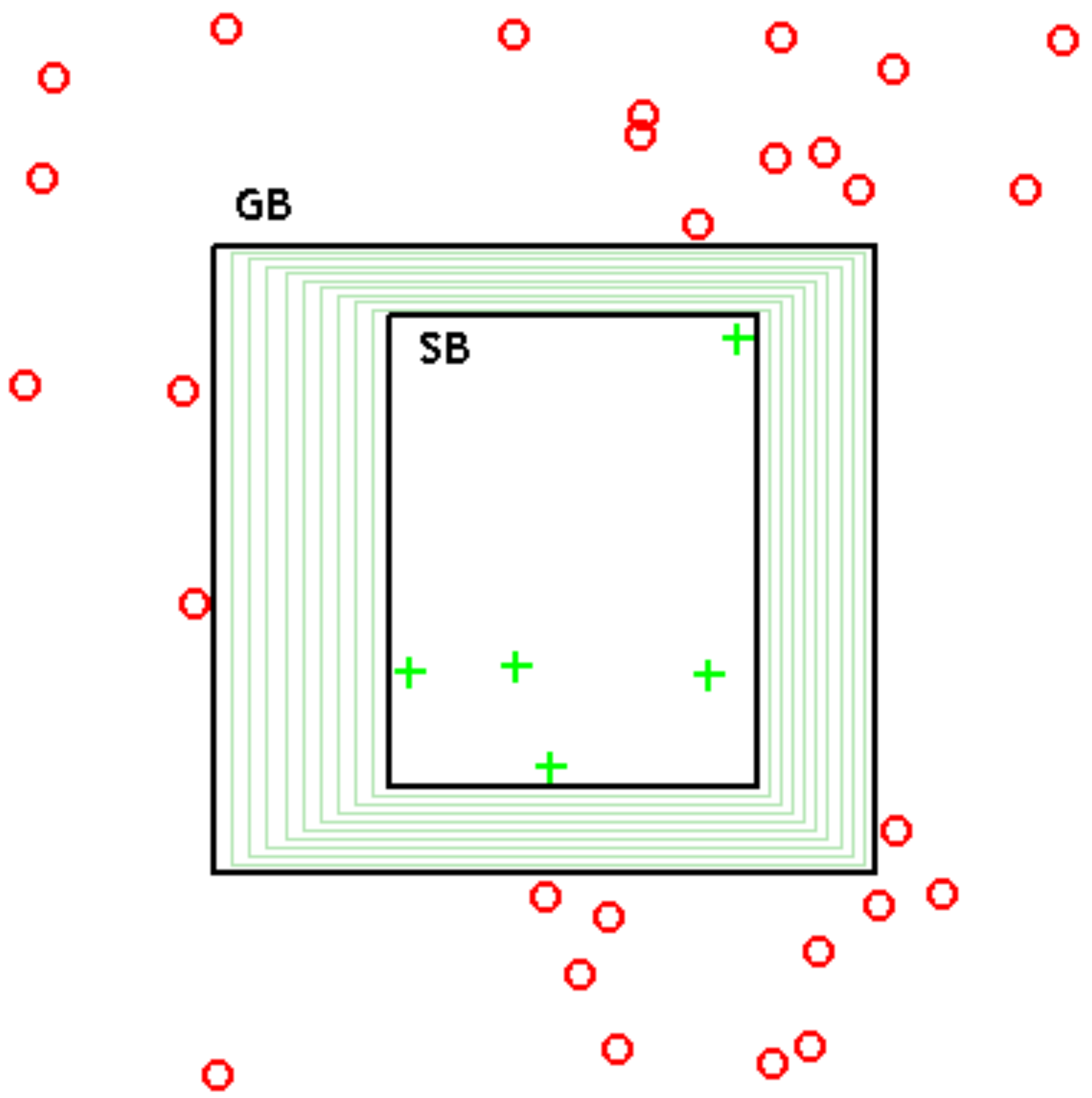


Figure from https://en.wikipedia.org/wiki/File:Version_space.png in the public domain

Version space (Mitchel 1978)

- Assume that there exist hypotheses consistent with the labeled data points $h : \mathscr{X} \rightarrow \mathscr{Y}$

  version space: $VS(D) = \{h \in H \,|\, \texttt{cons}(h, D)\}$

- **Specific hypotheses**: cover positive examples & as little remaining feature space
- **General hypotheses**: cover positive examples & as much of the remaining feature space
- **Version space:** represented as green rectangles

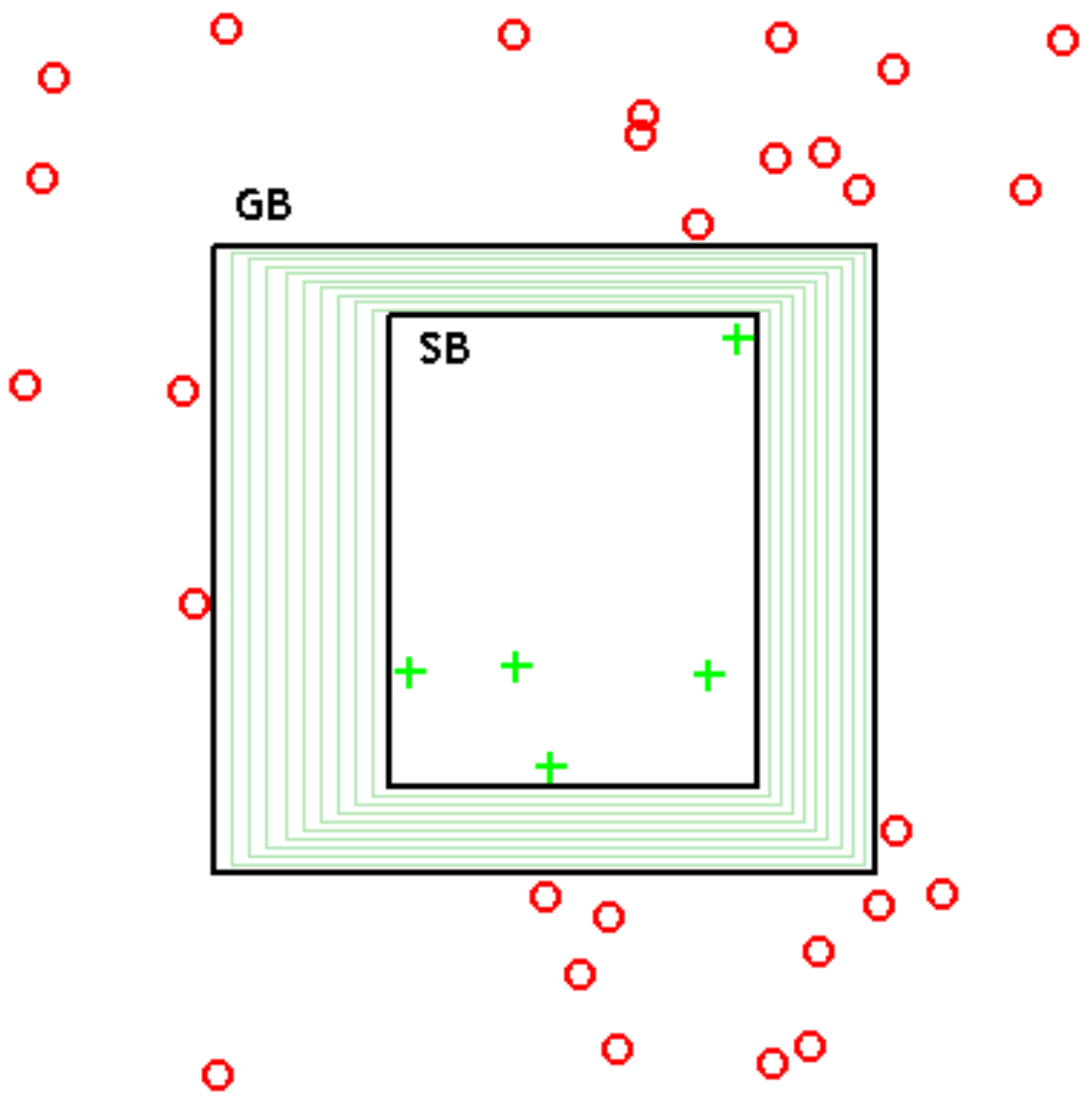


Figure from https://en.wikipedia.org/wiki/File:Version_space.png in the public domain

Version space reduction

“Generalization as Search”, Mitchell 1982

We could query such that the **version space**: $VS(D) = \{h \in H \,|\, \mathtt{cons}(h, D)\}$,i.e. the set of consistent hypotheses, quickly **gets reduced**

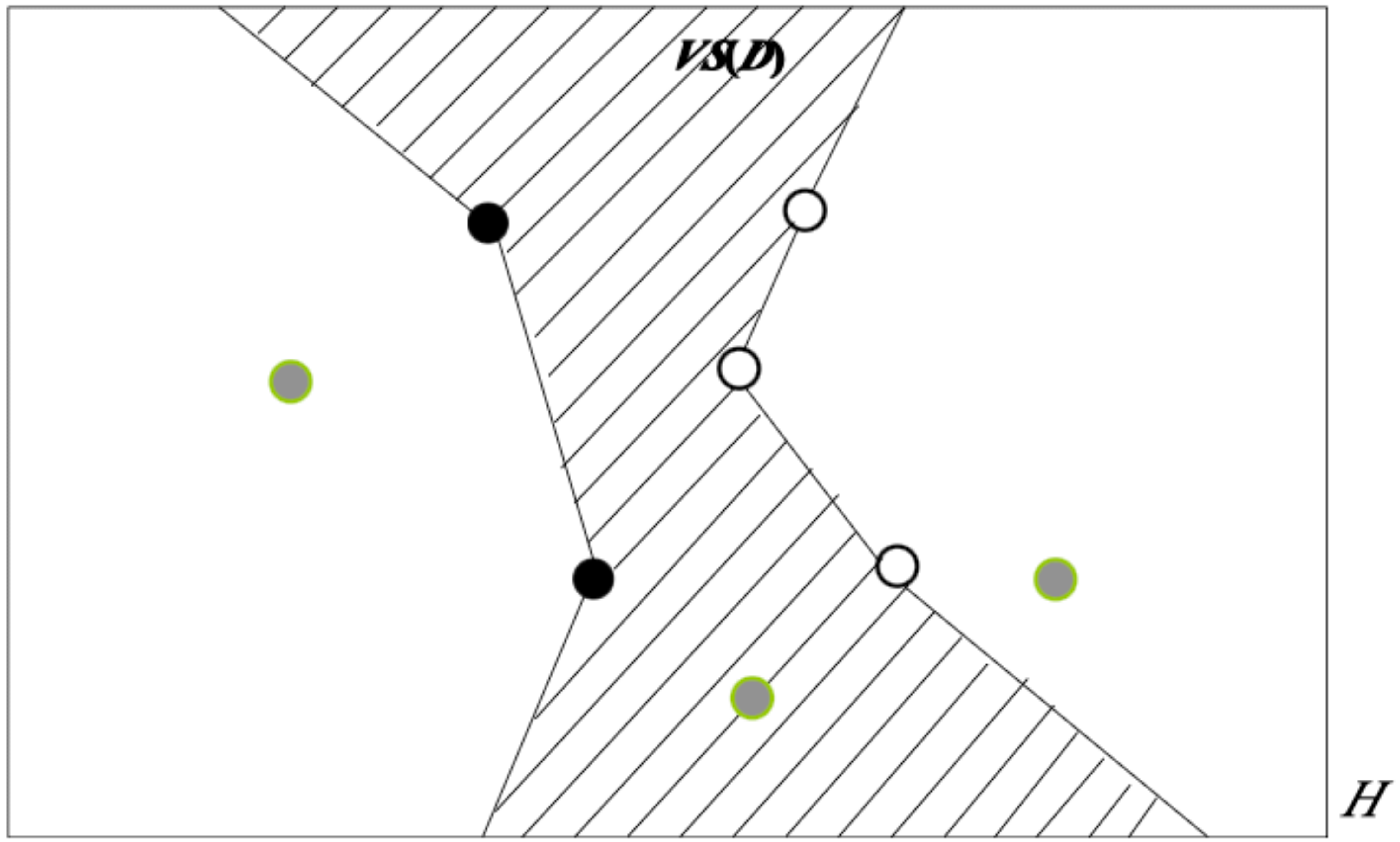
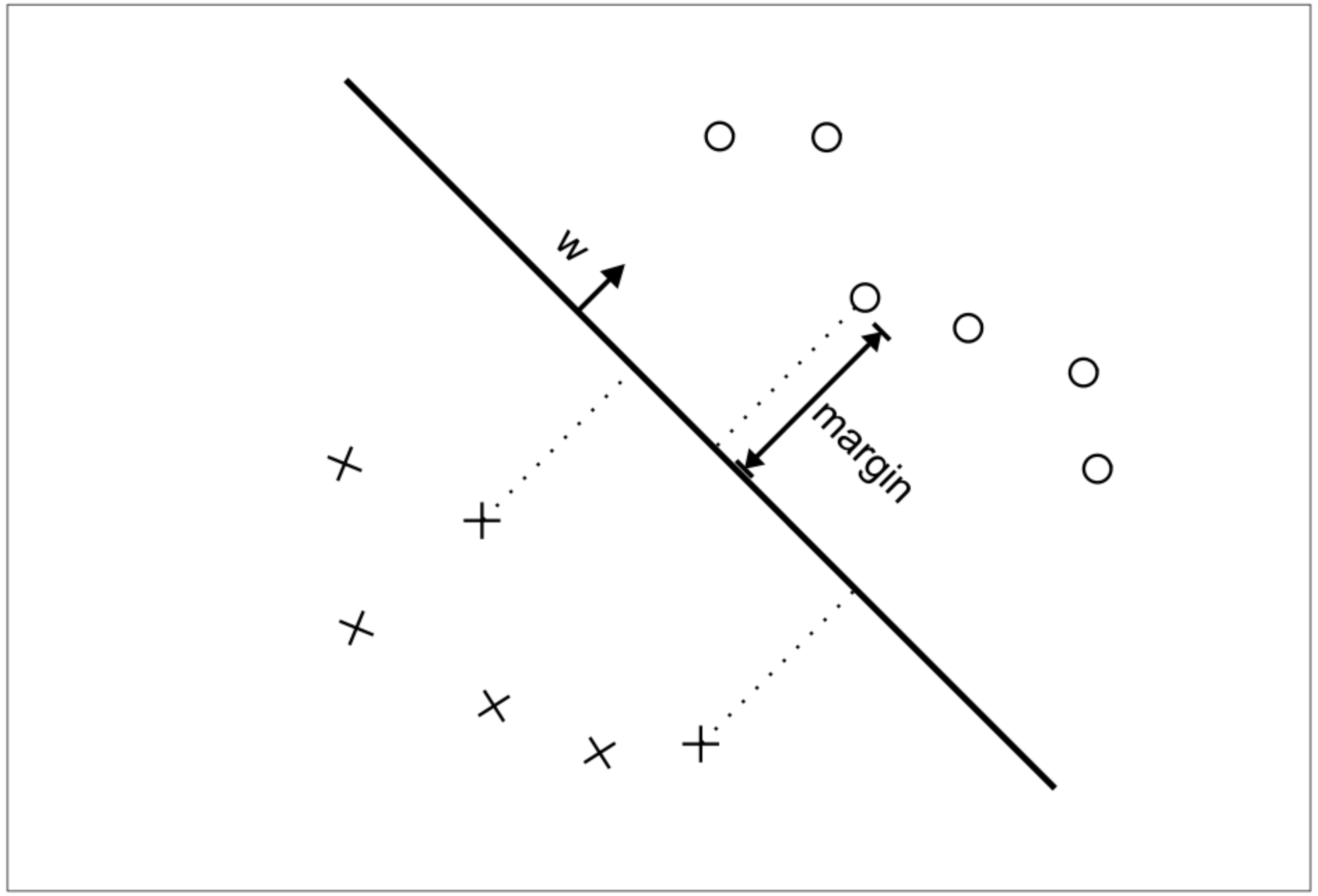


Figure from presentation of “Ensembles of Classifiers” by Evgueni Smirnov, slides available at: https://slideplayer.com/slide/10075963/

Active learning with support vector machines (SVM)

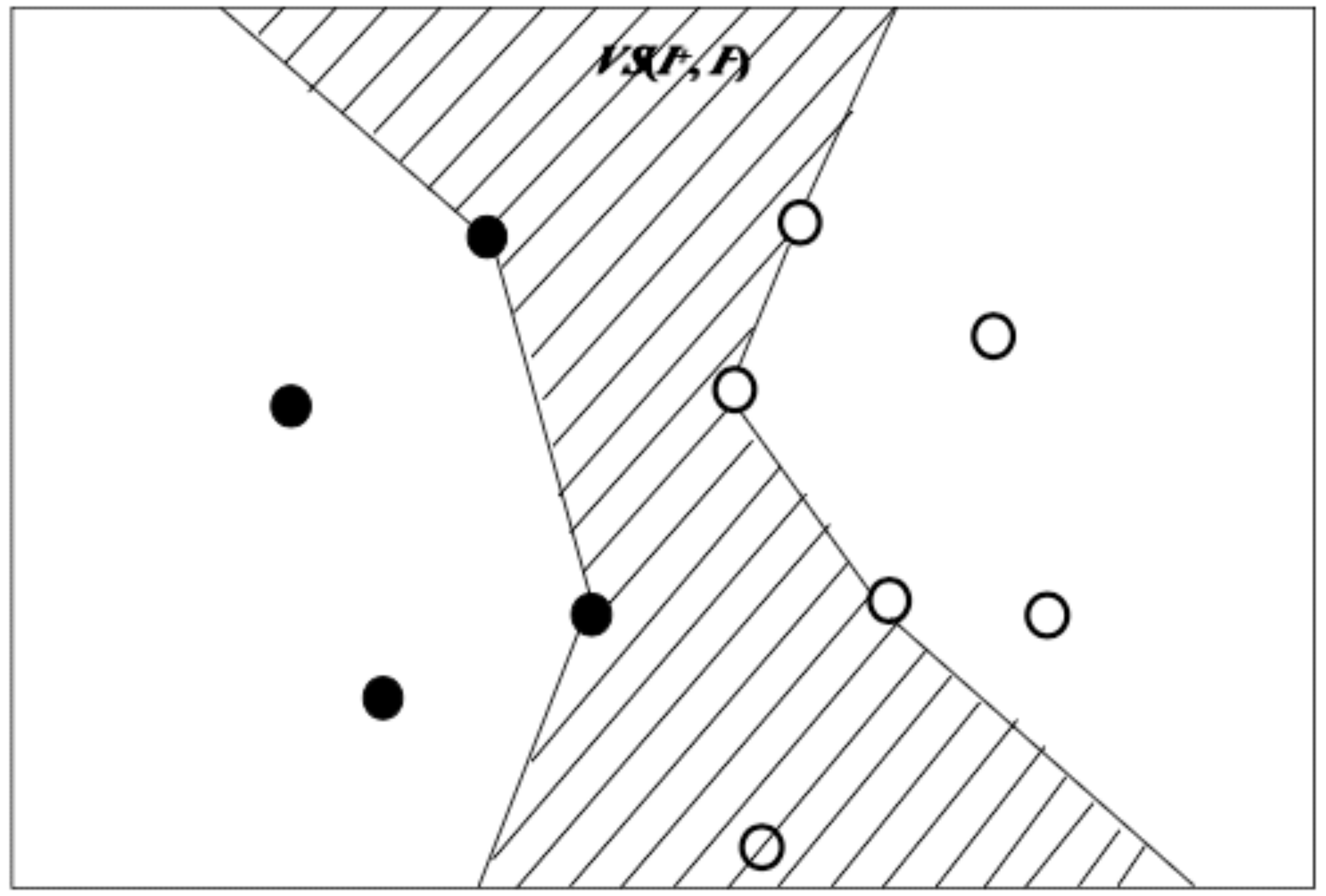
There are some models in which we can do this. Why?

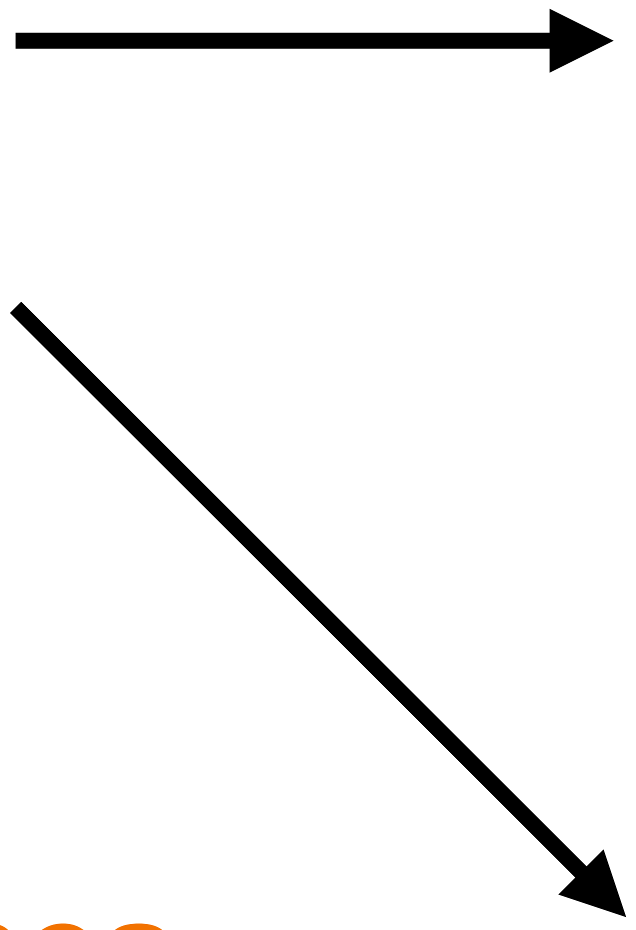
- Hyperplane chosen to **maximize margin to closest instances**: the support vectors

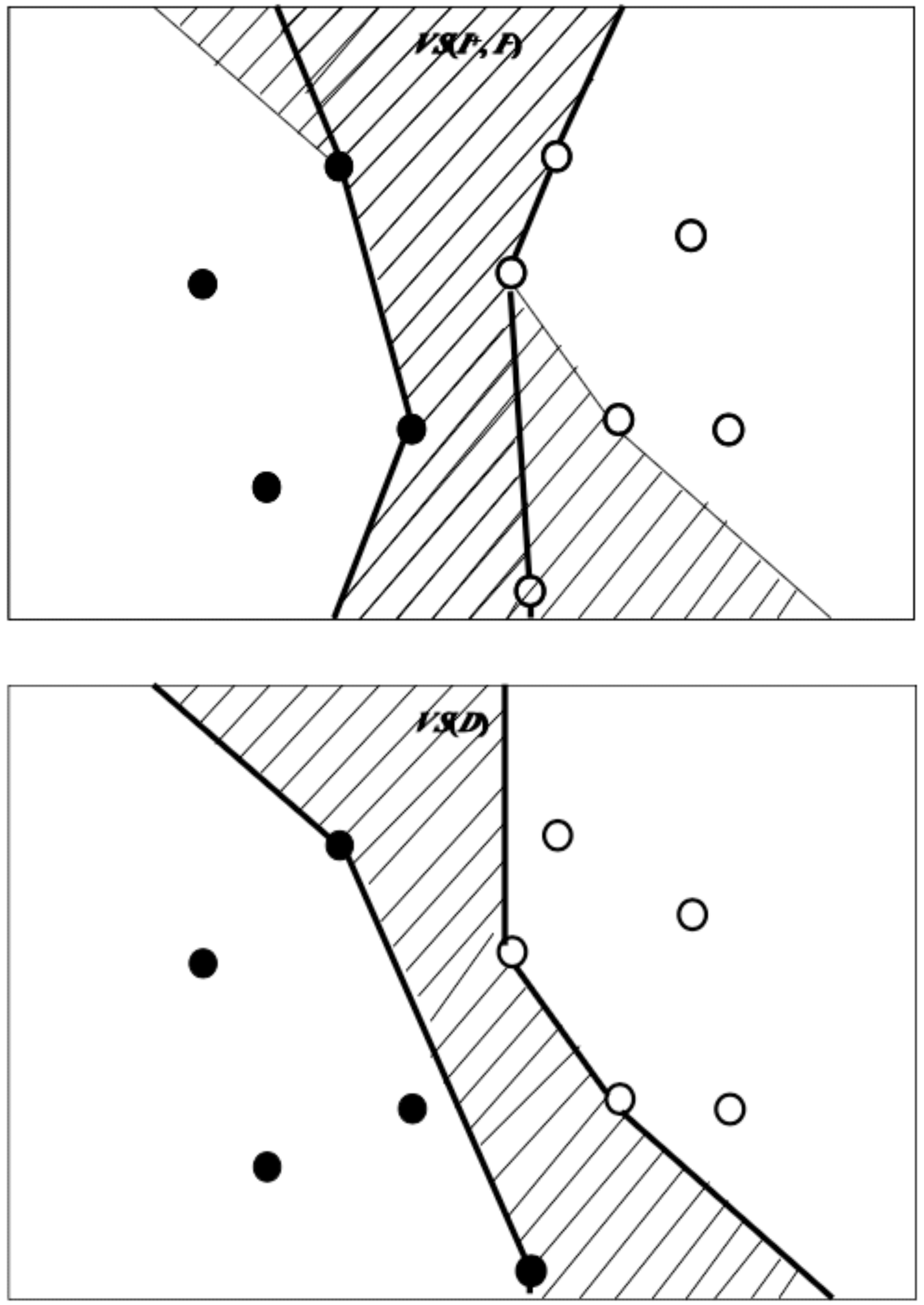


Tong & Koller, “Support Vector Machine Active Learning with Applications to Text Classification”, JMLR 2001

Active learning with SVM version space







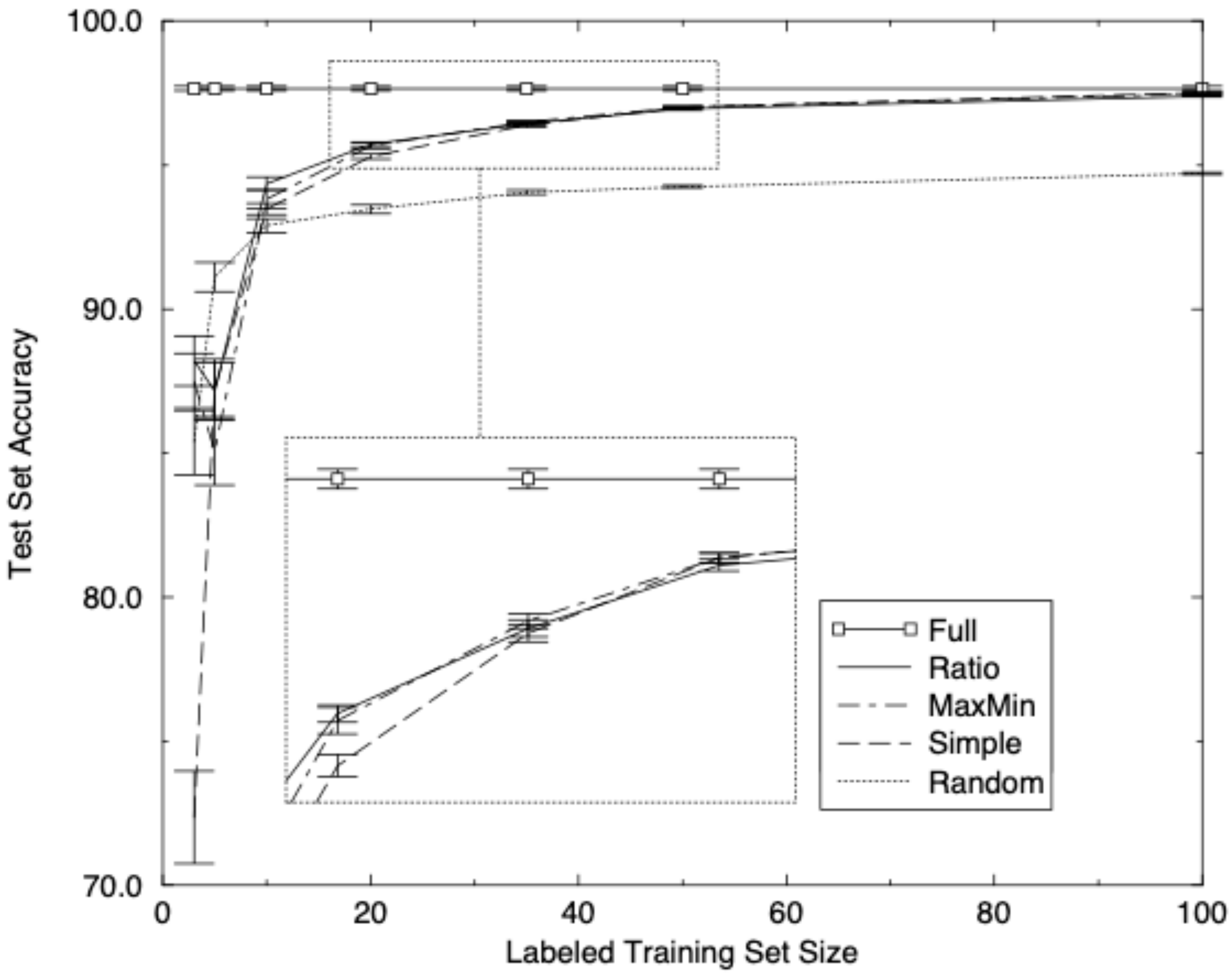
Version space is **set of hyperplanes**

(or could be redefined through vectors W)

Figure from presentation of “Ensembles of Classifiers” by Evgueni Smirnov, slides available at: https://slideplayer.com/slide/10075963/

Active learning with SVM version space

- Rapidly reduce version space
- Intuitively: choose queries that halve the version space
- Various approximations: is version space symmetric? Estimates of the size? etc.



Tong & Koller, "Support Vector Machine Active Learning with Applications to Text Classification", JMLR 2001

Reducing the set of consistent hypotheses does not regard the evaluation metric

An alternative to version space (the ML way)

We could also take a look at the machine learning **loss** and include points that would:

- **most reduce the expected error**
- **most change the current model**

An alternative to version space (the ML way)

We could also take a look at the machine learning **loss** and include points that would:

- **most reduce the expected error**
- **most change the current model**

"First-order Markov active learning aims to select a query $x^\star$, such that when the query is given label $y^\star$ and added to the training set, the learner trained on the resulting set $D+(x^\star,y^\star)$ has lower error than any other x"

Roy & McCallum, "Toward Optimal Active Learning through Monte Carlo Estimation of Error Reduction", ICML 2001)

(See also Cohn et al, "Active learning with statistical models", JAIR 4, 1996)

The simplest (?) approach

Version spaces & expected error reduction are hard (& heavy to compute).

Simple heuristics are thus popular

The simplest (?) approach

Version spaces & expected error reduction are hard (& heavy to compute).

Simple heuristics are thus popular, but have lots of caveats (tomorrow)

1. Create an initial classifier
2. While teacher is willing to label examples
   (a) Apply the current classifier to each unlabeled example
   (b) Find the b examples for which the classifier is least certain of class membership
   (c) Have the teacher label the subsample of b examples
   (d) Train a new classifier on all labeled examples

Lewis & Gale, “A Sequential Algorithm for Training Text Classifiers”, ACM-SIGIR conference on research and development in information retrieval 1994

Query by committee

We could maximize information gain between multiple models: **ensembles**

Query by committee

We could maximize information gain between multiple models: **ensembles**

Query by a committee of two

Repeat the following until n queries have been accepted

1. Draw an unlabeled input $x \in X$ at random from $\mathcal{D}$.
2. Select two hypotheses h_1, h_2 from the posterior distribution. In other words, pick two hypotheses that are consistent with the labeled examples seen so far.
3. If $h_1(x) \neq h_2(x)$ then query the teacher for the label of x, and add it to the training set.

Seung et al, "Query by Committee", COLT 1992, and Freund, Seung et al, "Information, Prediction, and Query by Committee", NeurIPS 1992

Query by committee

We could maximize information gain between multiple models: **ensembles**

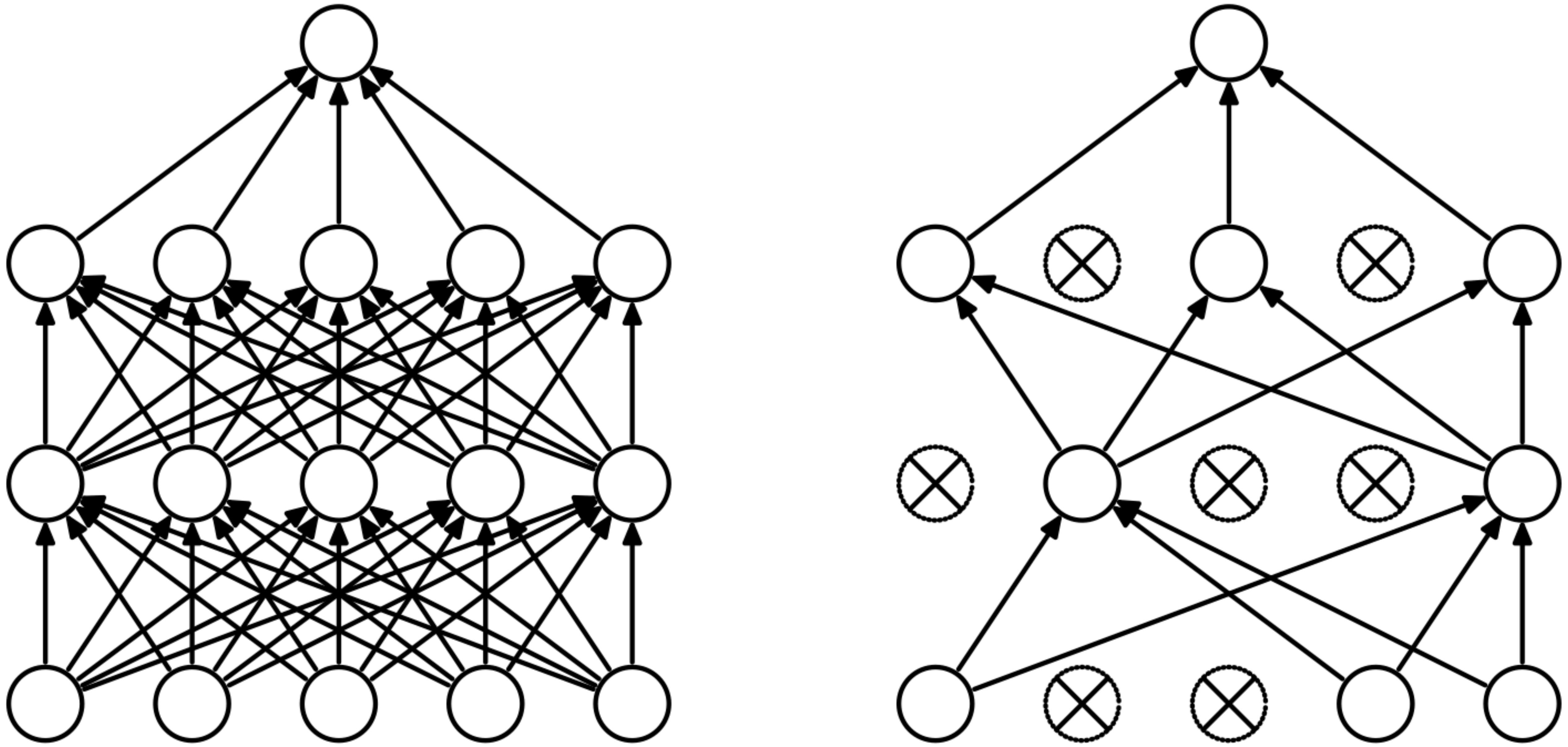
Query by a committee of two
Repeat the following until n queries have been accepted

1. Draw an unlabeled input $x \in X$ at random from $\mathcal{D}$.
2. Select two hypotheses h_1, h_2 from the posterior distribution. In other words, pick two hypotheses that are consistent with the labeled examples seen so far.
3. If $h_1(x) \neq h_2(x)$ then query the teacher for the label of x, and add it to the training set.

Could also be interpreted as reducing the version space across models

Seung et al, “Query by Committee”, COLT 1992, and Freund, Seung et al, “Information, Prediction, and Query by Committee”, NeurIPS 1992

Monte Carlo Dropout (Gal et al, ICML 2016)



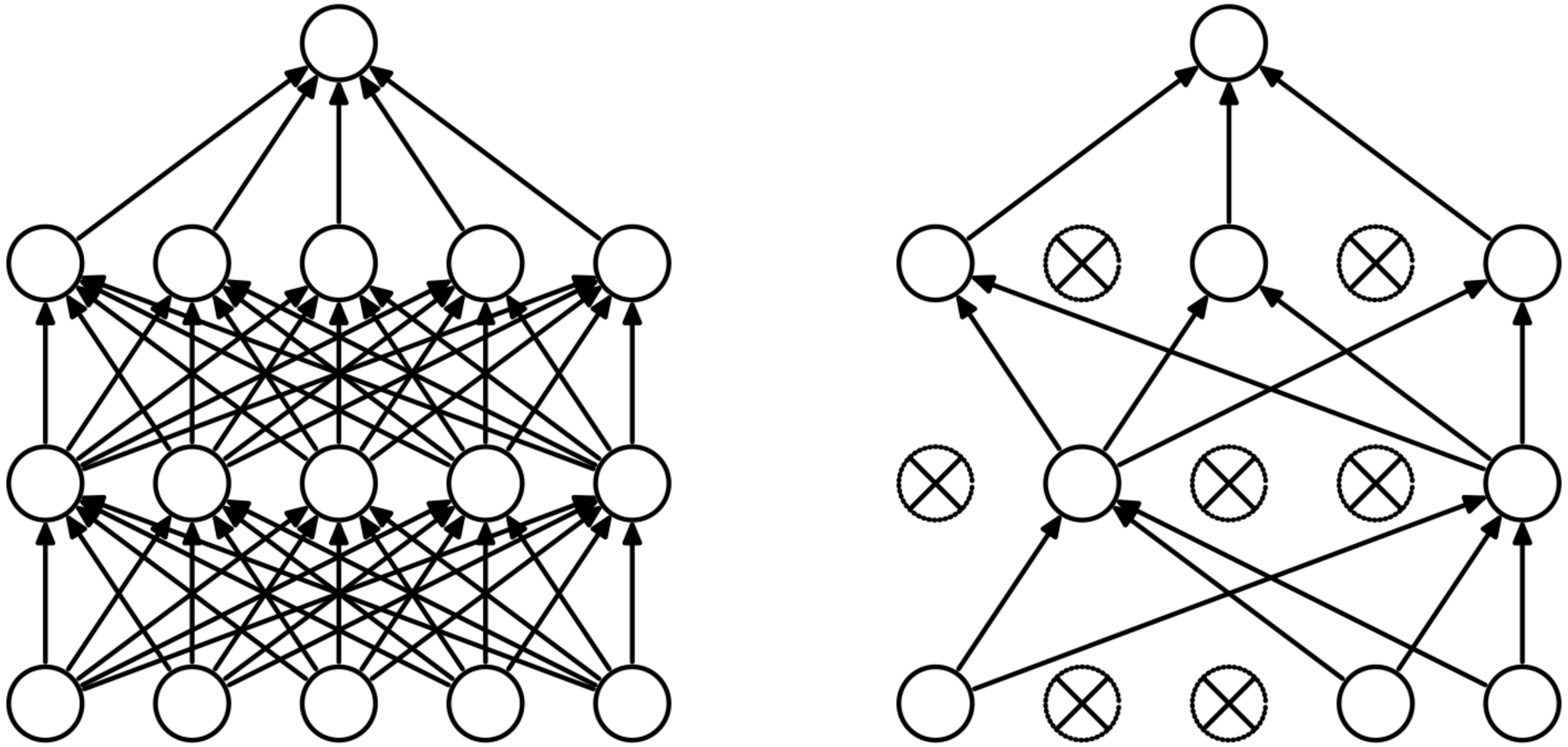
(a) Standard Neural Net

(b) After applying dropout.

- Make use of dropout: randomly turning off units in a model

Srivastava et al, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", JMLR 15, 2014

Monte Carlo Dropout (Gal et al, ICML 2016)



(a) Standard Neural Net

(b) After applying dropout.

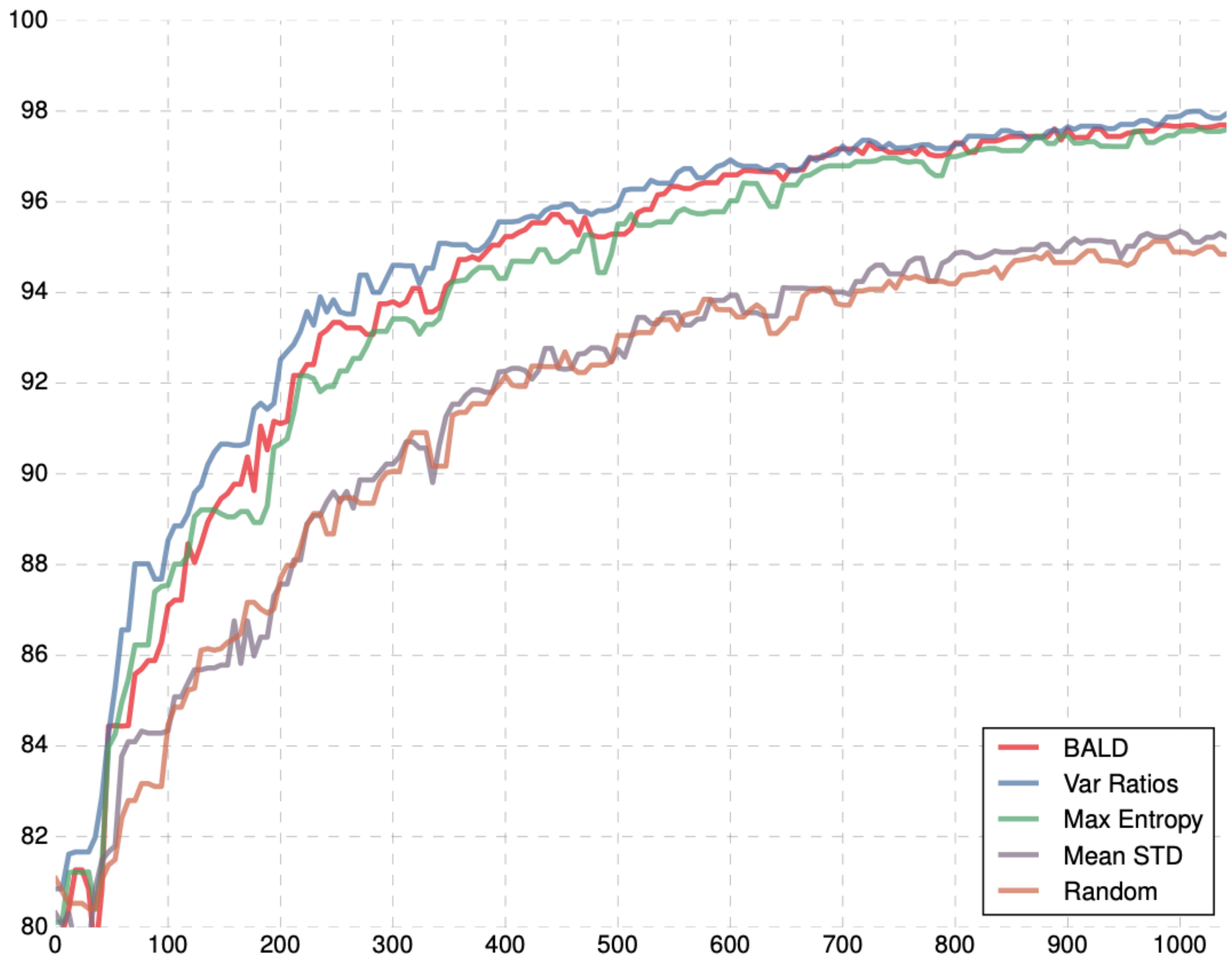
- Make use of dropout: randomly turning off units in a model
- Bayesian interpretation: Bernoulli distribution on the parameters
- Stochastic forward passes to get variation in predictions

Srivastava et al, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”, JMLR 15, 2014

Monte Carlo Dropout (Gal et al, ICML 2016)

MCD could be useful as an approximation to using multiple model based ensembles

The acquisition function can still be entropy, standard deviation in output confidence etc.



Gal et al, “Deep Bayesian Active Learning with Image Data”, ICML 2017

Random is hard to beat.
Why aren't these approaches a lot better?

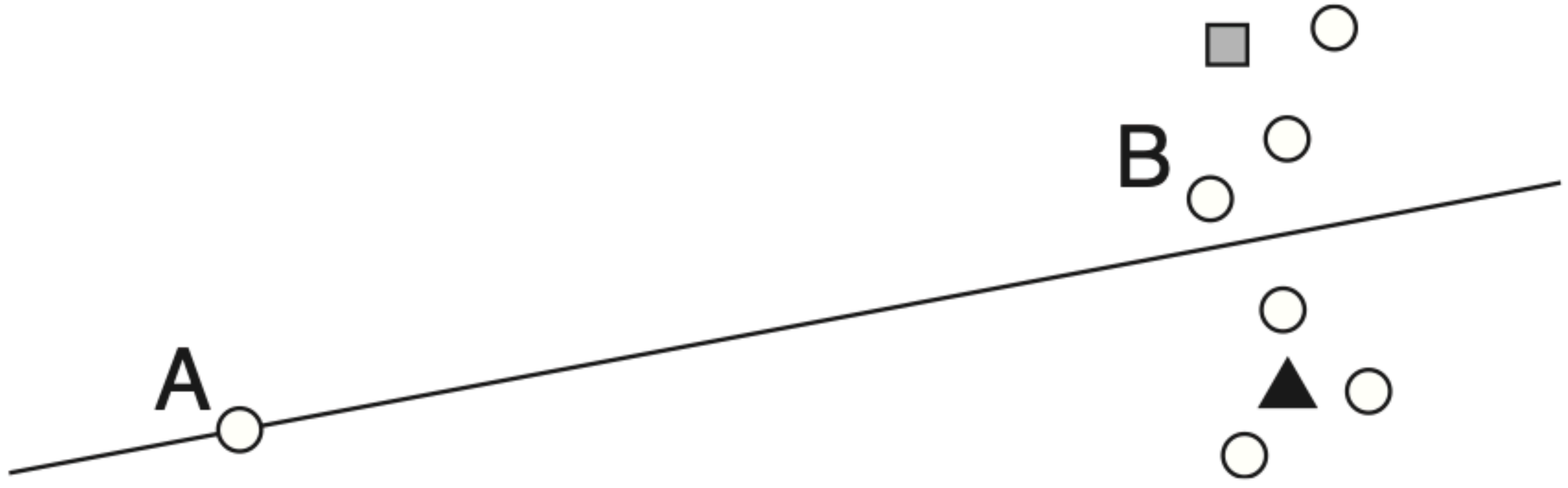


Figure 2: An illustration of when uncertainty sampling can be a poor strategy for classification. Shaded polygons represent labeled instances ($\mathcal{L}$), and circles represent unlabeled instances ($\mathcal{U}$). Since A is on the decision boundary, it will be queried as the most uncertain. However, querying B is likely to result in more information about the data as a whole.

Settles & Craven, "An Analysis of Active Learning Strategies for Sequence Labeling Tasks", EMNLP 2008

There are more challenges with data that is “far away” (tomorrow). Let us first complete the picture

Active learning perspectives

Version space reduction

reduce the set/space of possible hypotheses $h : \mathscr{X} \rightarrow \mathscr{Y}$ by removing the ones that are inconsistent with the data

Uncertainty & heuristics

use the predictions, or maybe even better, uncertainty in the predictions for the queries

Core sets & representation learning

Maximize distribution coverage instead of reducing the possible set of hypotheses

Representations & core sets

What if we allow to use & even train on the unlabelled pool?

Assumption: a **"teacher" information source is allowed**, e.g. generative model

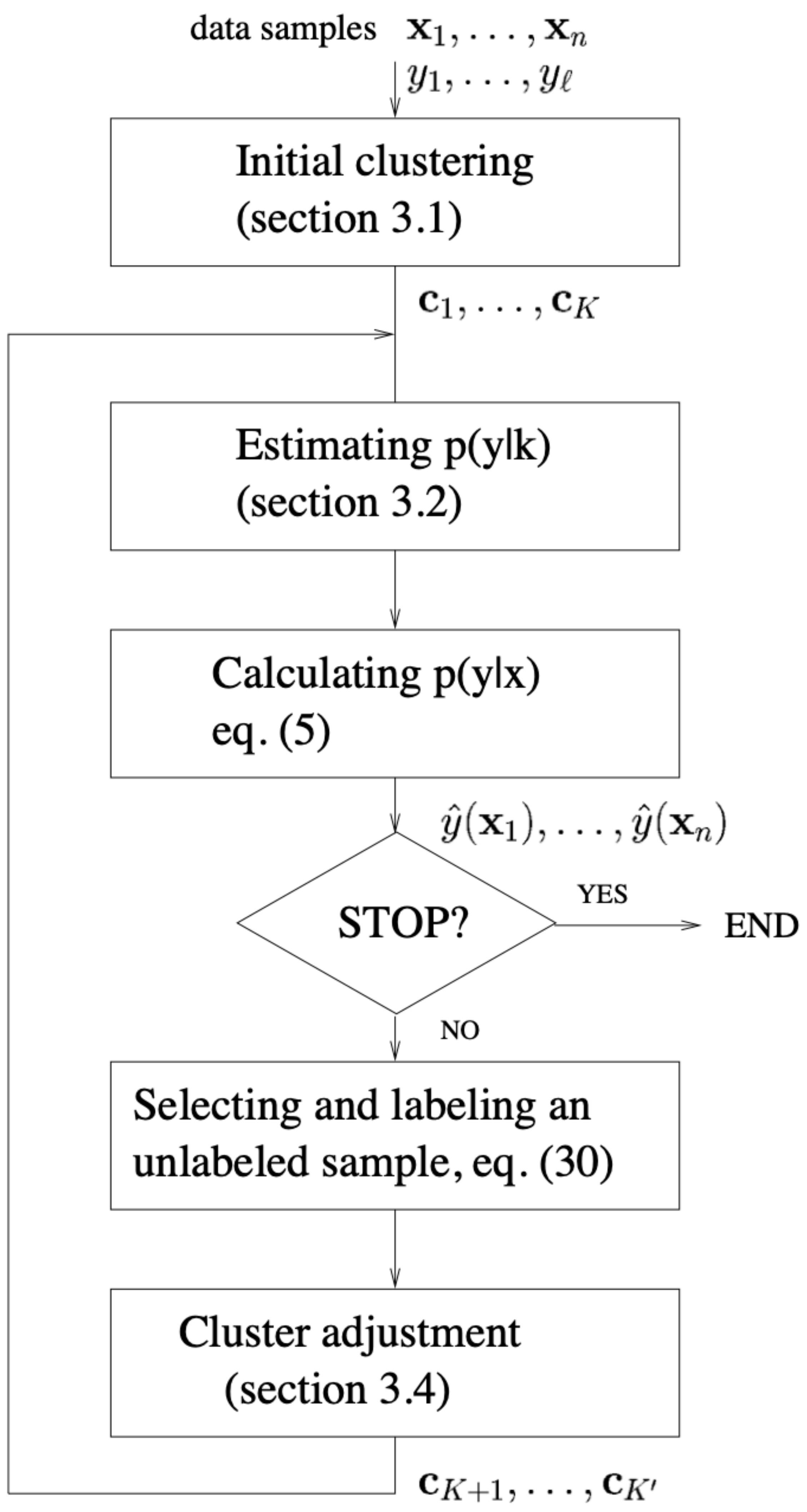
We wouldn't necessarily get a lot of advantage of generative models in active learning, unless **we also train on the (unlabelled pool)**

We could then also **make use of core sets**, as discussed for memory

Representations & core sets

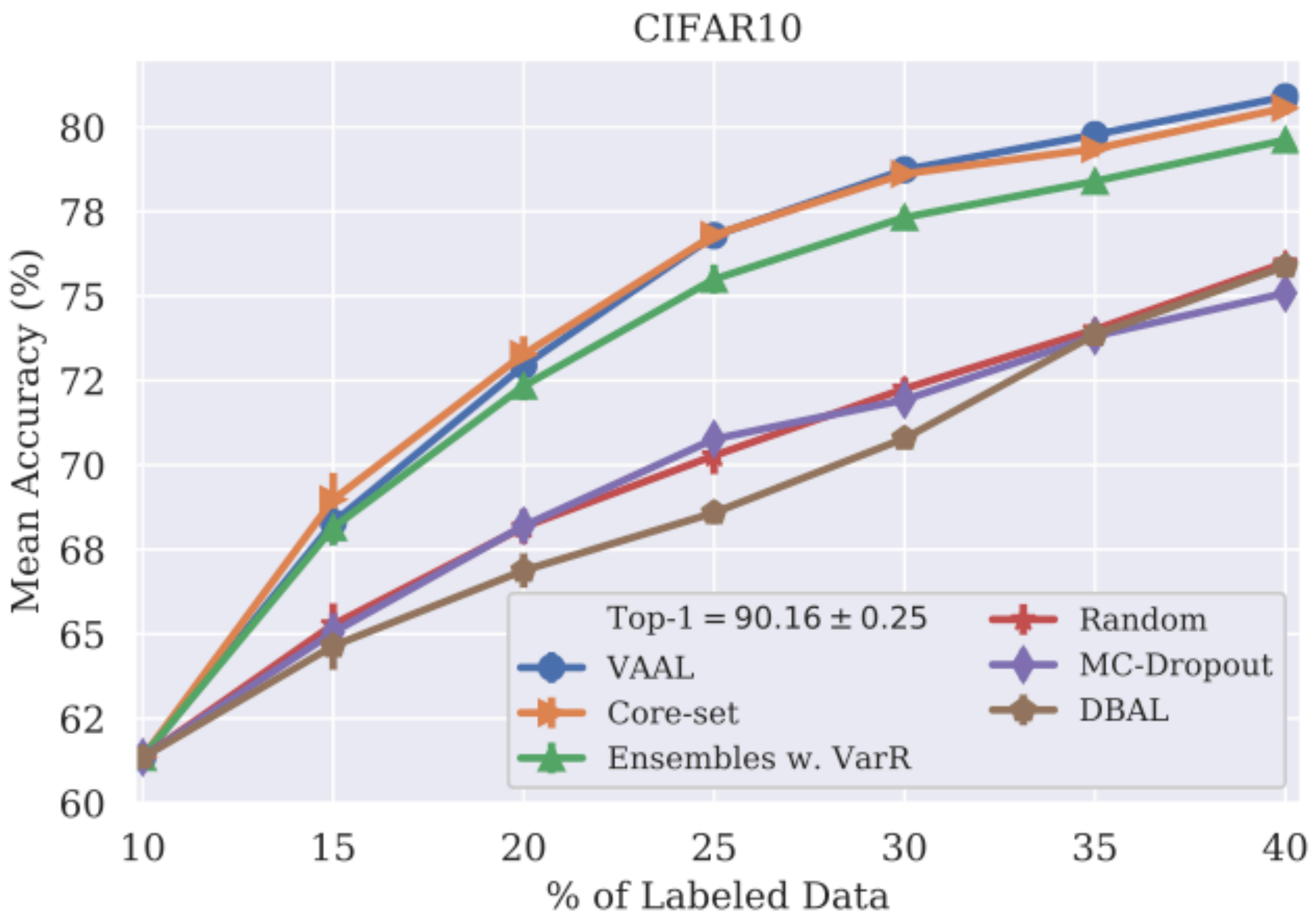
We could now try to:

- Pre-cluster our unlabelled data pool
- Compute core sets of the unlabelled data pool
- Learn a generative model & representations on the unlabelled data pool



H.T. Nguyen et al, “Active Learning Using Pre-clustering”, ICML 2004

Representations & core sets



Sinha et al, “Variational Adversarial Active Learning”, ICCV 2019

Intermediate summary: assumptions & trade-offs

Intermediate summary: active learning perspectives

Version space reduction *(Hypotheses)*

reduce the set/space of possible hypotheses $h : \mathscr{X} \rightarrow \mathscr{Y}$ by removing the ones that are inconsistent with the data

Uncertainty & heuristics *(Novelty)*

use the predictions, or maybe even better, uncertainty in the predictions for the queries

Core sets & representation learning - accessing the entire pool *(Diversity)*

maximize distribution coverage instead of reducing the possible set of hypotheses

Intermediate summary: active learning perspectives

Techniques

- Version space reduction

& (some of) their assumptions

- Set of hypotheses is clear

Intermediate summary: active learning perspectives

Techniques

- Version space reduction
- Minimum confidence
- Maximum entropy

& (some of) their assumptions

- Set of hypotheses is clear
- No overconfidence phenomenon and out-of-distribution/task data

Intermediate summary: active learning perspectives

Techniques	**& (some of) their assumptions**
• Version space reduction	• Set of hypotheses is clear
• Minimum confidence • Maximum entropy	• No overconfidence phenomenon and out-of-distribution/task data
• Model “uncertainty” (output variability)	• Accurate uncertainty everywhere
• Ensembles/query by committee	• Training of multiple models

Intermediate summary: active learning perspectives

Techniques

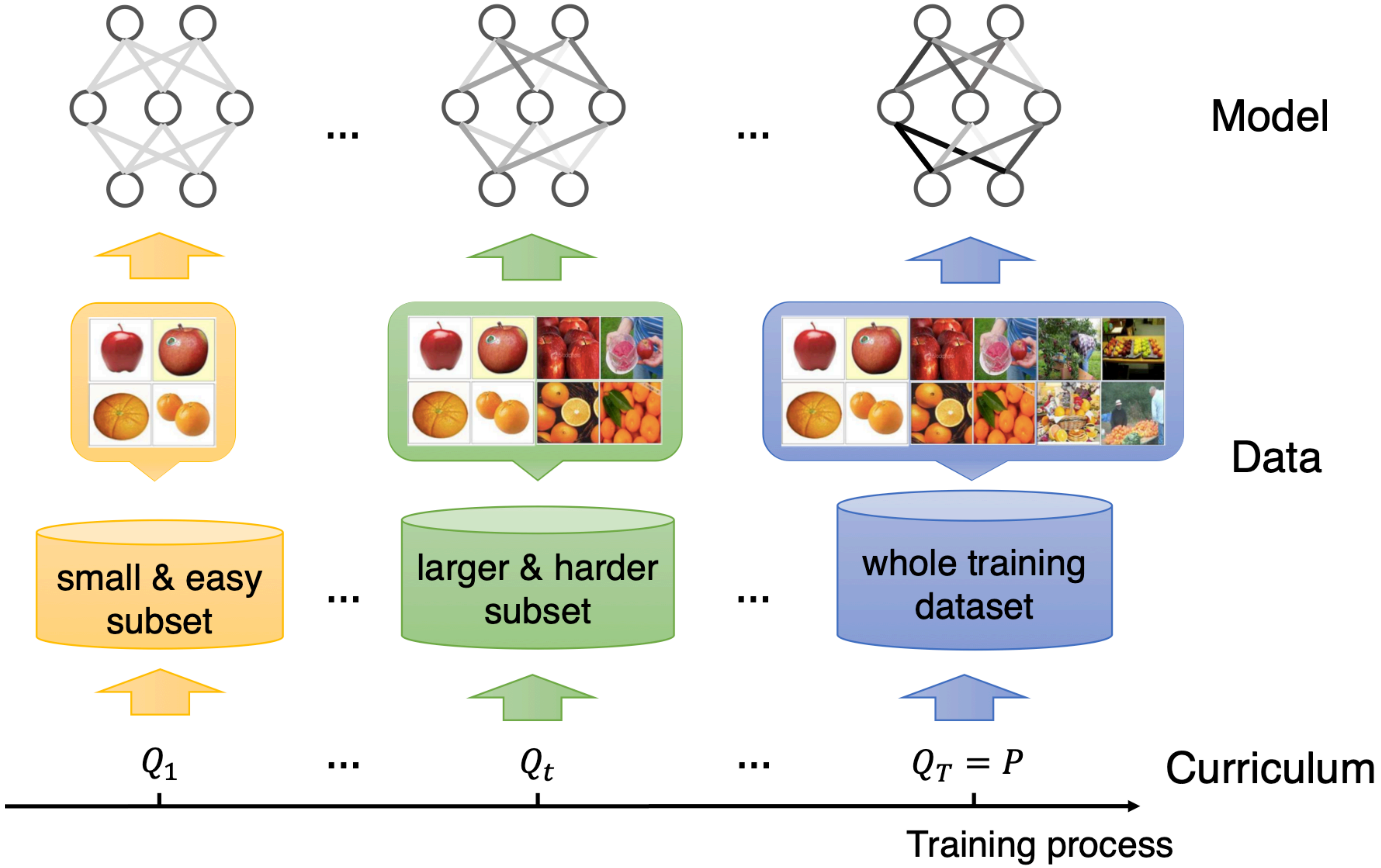
- Version space reduction
- Minimum confidence
- Maximum entropy
- Model "uncertainty" (output variability)
- Ensembles/query by committee
- Representation learning on the pool
- Core sets

& (some of) their assumptions

- Set of hypotheses is clear
- No overconfidence phenomenon and out-of-distribution/task data
- Accurate uncertainty everywhere
- Training of multiple models
- Upfront training of entire pool (no data stream) (access + computational expense)

There is another aspect to consider:
the informativeness of data + difficulty to learn

Curriculum Learning



what is "**easy**" & what is a "**harder**" subset/dataset?

And what is the difference to **informativeness**?

Wang et al, "A Survey on Curriculum Learning", TPAMI 2021

Let's start with an intuitive example: Ranking language model trained with vs without curriculum on Wikipedia

"Error" is log of the rank of the next word (within 20k-word vocabulary).

Let’s start with an intuitive example: Ranking language model trained with vs without curriculum on Wikipedia

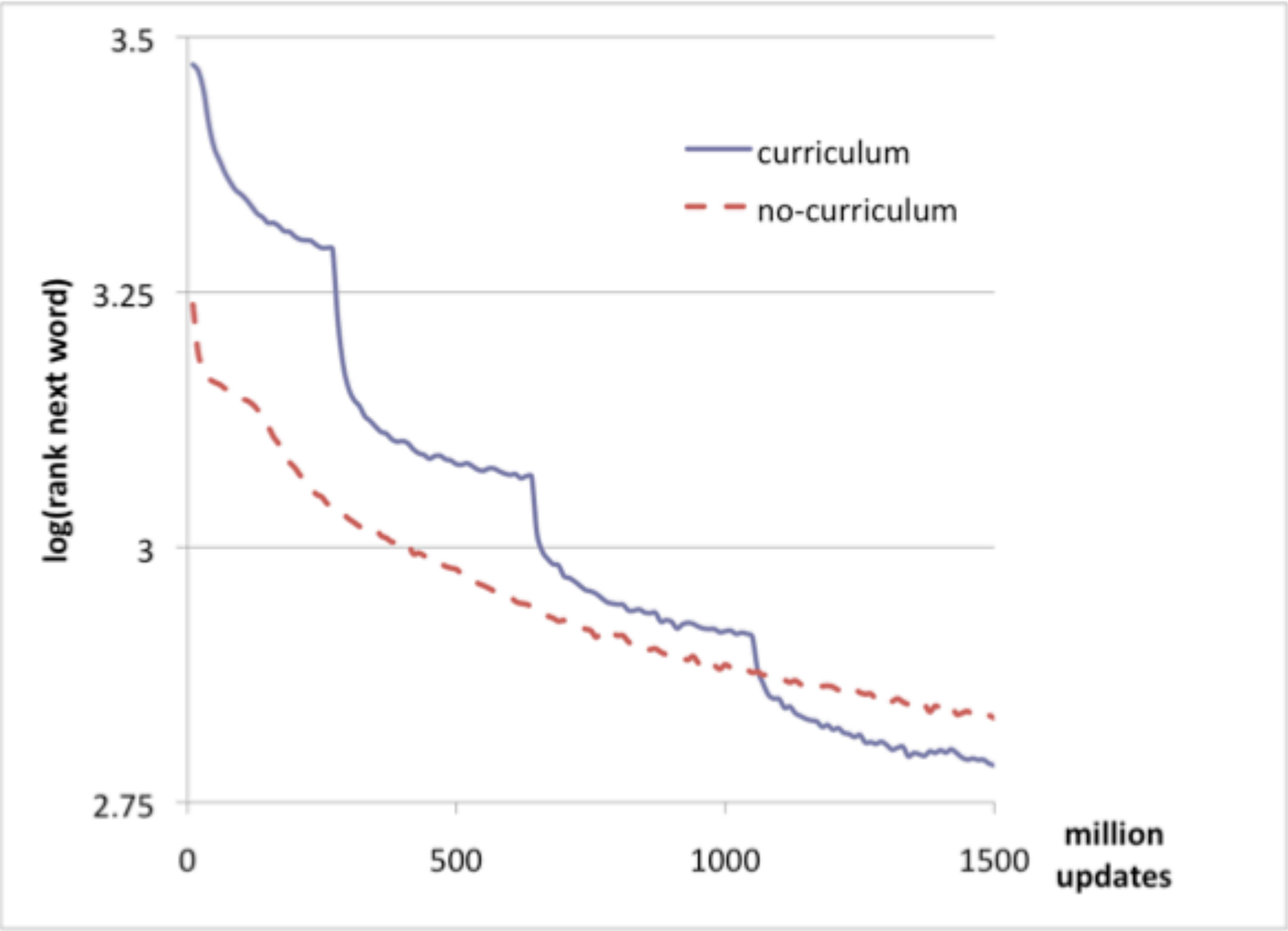
“Error” is log of the rank of the next word (within 20k-word vocabulary).

1. The curriculum-trained model skips examples with words outside of 5k most frequent words
2. Then skips examples outside 10k most frequent words and so on

Let’s start with an intuitive example: Ranking language model trained with vs without curriculum on Wikipedia

“Error” is log of the rank of the next word (within 20k-word vocabulary).

1. The curriculum-trained model skips examples with words outside of 5k most frequent words
2. Then skips examples outside 10k most frequent words and so on



Bengio et al, “Curriculum Learning”, ICML 2009

Curriculum learning: the two key challenges

Scoring function (or difficulty measurer):

Any function that provides us with an estimate of the difficulty of the instances in our dataset(s).

Curriculum learning: the two key challenges

Scoring function (or difficulty measurer):

Any function that provides us with an estimate of the difficulty of the instances in our dataset(s).

Pacing function (or training scheduler):

(sometimes also called competence, as we'll see)

The function that tells us how to interleave samples into the training process over time.

Curriculum Learning

Definition 1: Original Curriculum Learning [6]. A curriculum is a sequence of training criteria over T training steps: $\mathcal{C} = \langle Q_1, \ldots, Q_t, \ldots, Q_T \rangle$. Each criterion Q_t is a reweighting of the target training distribution $P(z)$:

$$Q_t(z) \propto W_t(z) P(z) \quad \forall \text{example } z \in \text{training set } D, \tag{1}$$

such that the following three conditions are satisfied:

1) The entropy of distributions gradually increases, i.e., $H(Q_t) < H(Q_{t+1})$.
2) The weight for any example increases, i.e., $W_t(z) \leq W_{t+1}(z) \quad \forall z \in D$.
3) $Q_T(z) = P(z)$.

From Wang et al, “A Survey on Curriculum Learning”, TPAMI 2021, based on original definition by Bengio et al, “Curriculum Learning”, ICML 2009

Curriculum Learning

Curriculum learning: the more intuitive definition

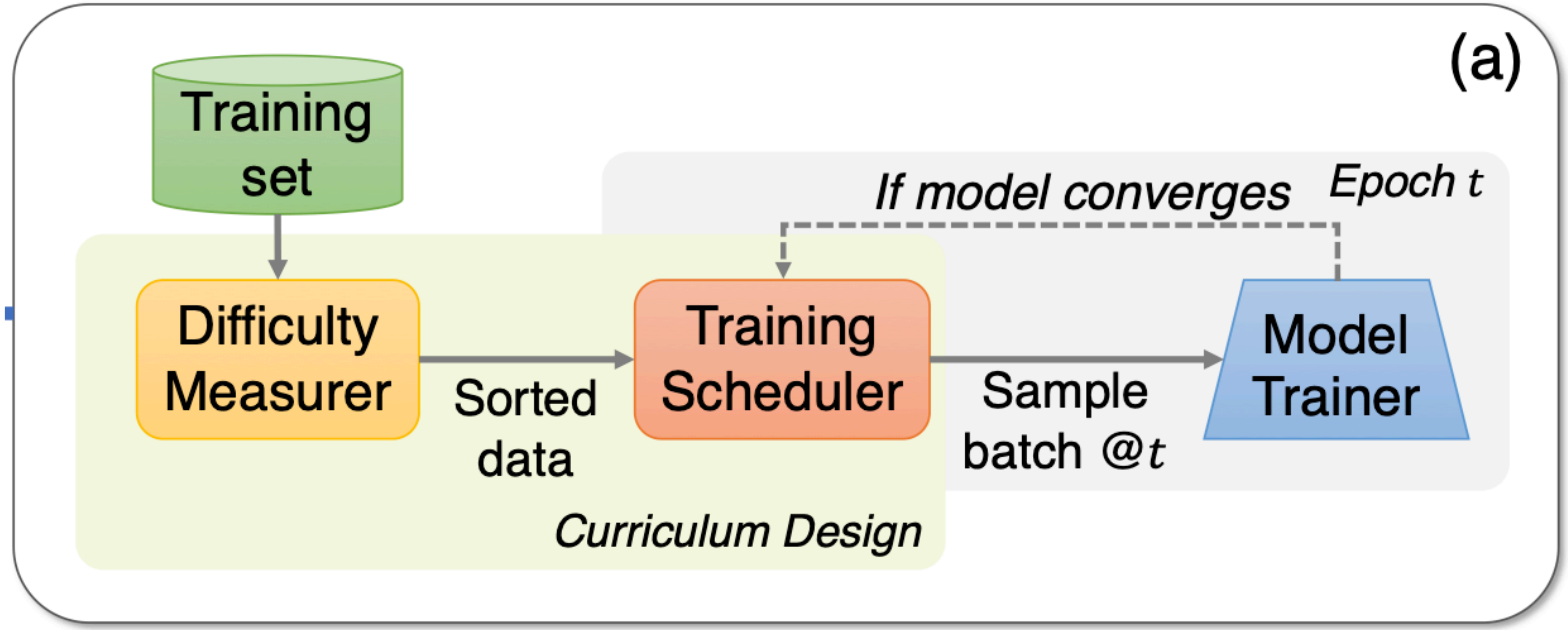
(with a little bit of a tautology)

Definition 3: Generalized Curriculum Learning. Discarding the definition of Q_t (Eq. 1) and its three conditions in Definition 1, a curriculum is a sequence of training criteria over T training steps. Each criterion Q_t includes the design for all the elements in training a machine learning model, e.g., data/tasks, model capacity, learning objective, etc. Curriculum learning is the strategy that trains a model with such a curriculum.

From Wang et al, "A Survey on Curriculum Learning", TPAMI 2021, based on original definition by Bengio et al, "Curriculum Learning", ICML 2009

Curriculum learning

Let's start by considering a pre-defined curriculum, inspired by learning from "textbook style" content



Wang et al, "A Survey on Curriculum Learning", TPAMI 2021

Can you think of ways to define “difficulty”?

How to define difficulty: it is task & model specific

TABLE 2
Common types of predefined Difficulty Measurer. The "+" in $\propto$Easy means the higher the measured value, the easier the data example, and the "-" has the opposite meaning.

Difficulty Measurer*	Angle	Data Type	$\propto$Easy
Sentence length [86], [107]	Complexity	Text	-
Number of objects [122]	Complexity	Images	-
# conj. [50], #phrases [113]	Complexity	Text	-
Parse tree depth [113]	Complexity	Text	-
Nesting of operations [131]	Complexity	Programs	-
Shape variability [6]	Diversity	Images	-
Word rarity [50], [86]	Diversity	Text	-
POS entropy [113]	Diversity	Text	-
Mahalanobis distance [14]	Diversity	Tabular	-
Cluster density [11], [31]	Noise	Images	+
Data source [10]	Noise	Images	/
SNR / SND [7], [89]	Noise	Audio	-
Grammaticality [66]	Domain	Text	+
Prototypicality [113]	Domain	Text	+
Medical based [44]	Domain	X-ray film	/
Retrieval based [18], [82]	Domain	Retrieval	/
Intensity [30] / Severity [111]	Intensity	Images	+
Image difficulty score [106], [114]	Annotation	Images	-
Norm of word vector [68]	Multiple	Text	-

Wang et al, "A Survey on Curriculum Learning", TPAMI 2021

How to define difficulty: it is task & model specific

We have already seen that specific tasks allow for specific definitions of difficulty. Example: natural language translation (sentence length)

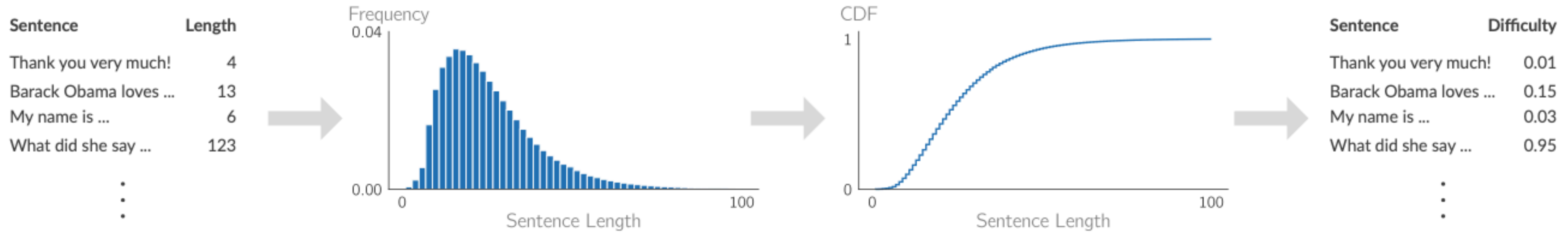


Figure 2: Example visualization of the preprocessing sequence used in the proposed algorithm. The histogram shown is that of sentence lengths from the WMT-16 En→De dataset used in our experiments. Here sentence lengths represent an example difficulty scoring function, d. "CDF" stands for the empirical "cumulative density function" obtained from the histogram on the left plot.

Platanios et al, "Competence based curriculum learning for neural machine translation", NAACL-HIT 2019

How to define difficulty: it is task & model specific

Another example: image segmentation (entropy/clutter)

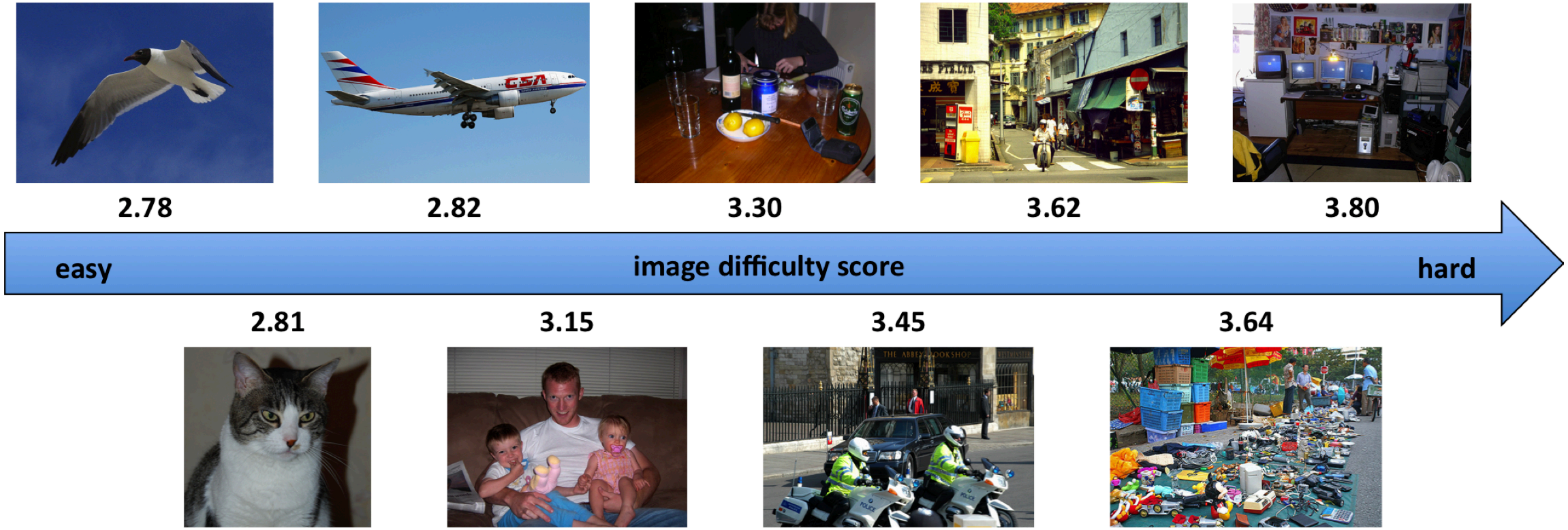


Figure 1. Images with difficulty scores predicted by our system in increasing order of their difficulty.

Ionescu et al, “How hard can it be? Estimating the difficulty of visual search in an image”, CVPR 2016

How to define difficulty: it is task & model specific

There are various dimensions to difficulty, not just (basic) data statistics.

Especially if we think about factors that relate to what humans find difficult

Compositional factors:

Size

“A sail boat on the ocean.”

Location

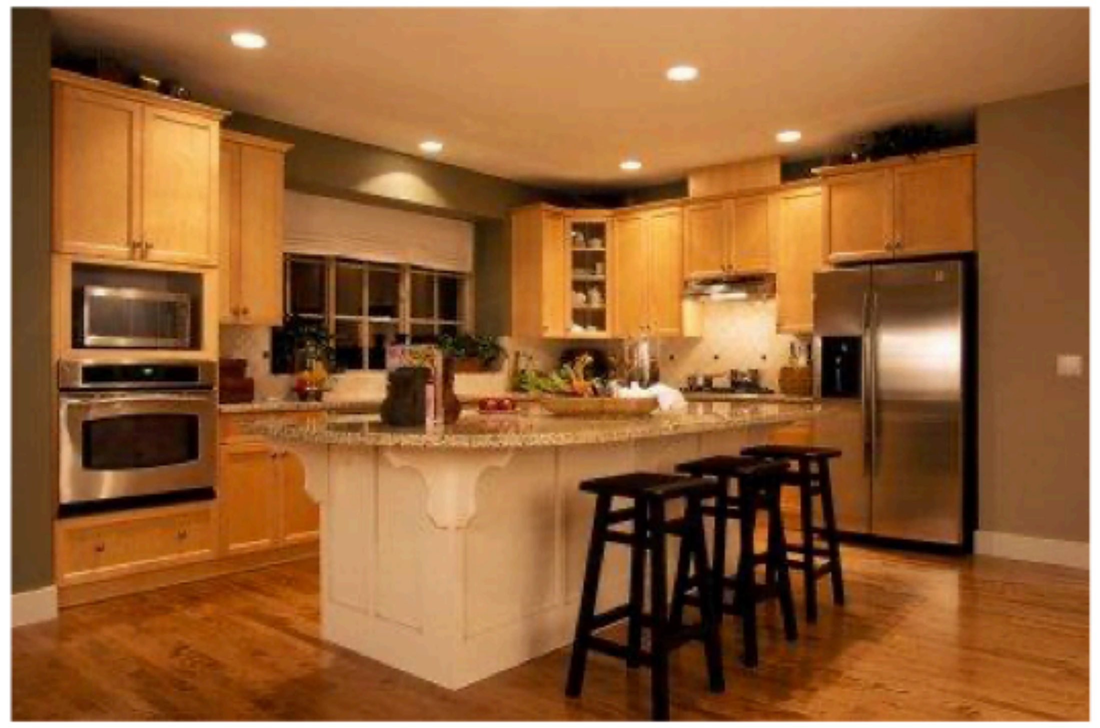
“Two men standing on beach.”

Semantic factors:

Object Type

“Girl in the street”

Scene Type & Depiction Strength



“kitchen in house”

Context factors:

Unusual object-scene Pair

“A tree in water and a boy with a beard”

Berg et al, “Understanding and predicting importance in images”, CVPR 2012

What is difficult for ML models?

But what is difficult for ML models & is this related to human perception?

Example: human response time

Collecting response times. We collected ground-truth difficulty annotations by human evaluators using the following protocol: (i) we ask each annotator a question of the type "Is there an {*object class*} in the next image?", where {*object class*} is one of the 20 classes included in the PASCAL VOC 2012; (ii) we show the image to the annotator; (iii) we record the time spent by the annotator to answer the question by "Yes" or "No". Finally, we use this response time to estimate the visual search difficulty.

Ionescu et al, "How hard can it be? Estimating the difficulty of visual search in an image", CVPR 2016

What is difficult for ML models?

Average human ranks about 80% image pairs in the same order as given by the mean response time of all annotators -> compared to Pascal "difficulty"

Collecting response times. We collected ground-truth difficulty annotations by human evaluators using the following protocol: (i) we ask each annotator a question of the type "Is there an $\{$*object class*$\}$ in the next image?", where $\{$*object class*$\}$ is one of the 20 classes included in the PASCAL VOC 2012; (ii) we show the image to the annotator; (iii) we record the time spent by the annotator to answer the question by "Yes" or "No". Finally, we use this response time to estimate the visual search difficulty.

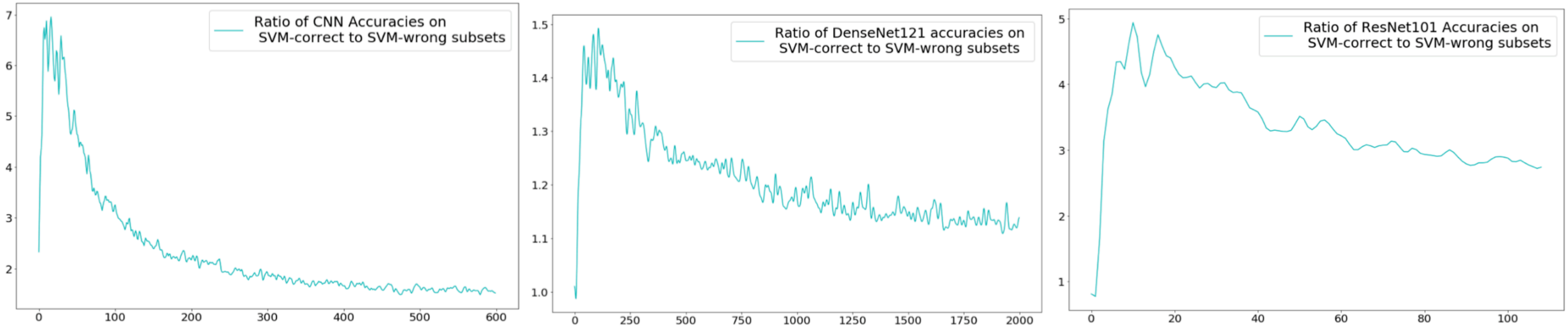
	Image property	Kendall τ
(i)	number of objects	0.32
(ii)	mean area covered by objects	-0.28
(iii)	non-centeredness	0.29
(iv)	number of different classes	0.33
(v)	number of truncated objects	0.22
(vi)	number of occluded objects	0.26
(vii)	number of difficult objects	0.20

Ionescu et al, "How hard can it be? Estimating the difficulty of visual search in an image", CVPR 2016

What is difficult for ML models?

Example: shallow embeddable examples seem to be learned first

A deep network in comparison to a SVM (random forest also in the paper)



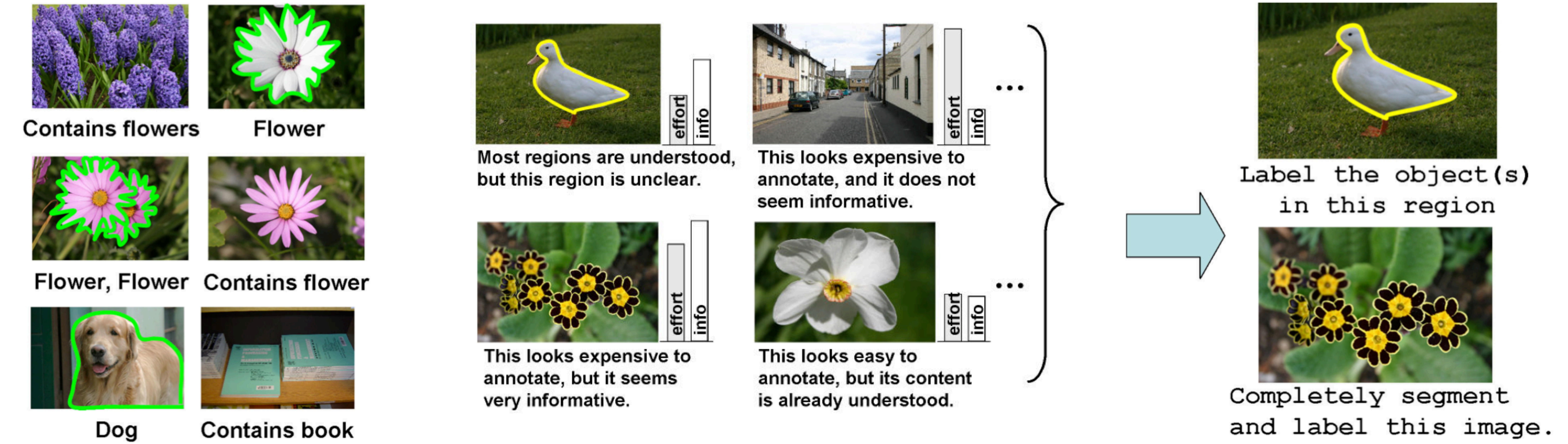
Ratio of Accuracies $\mathtt{R}^{\mathtt{i}}$ plotted against i with $\mathcal{M}$ being a Support Vector Machine.

Mangalam & Prabhu, "Do deep neural networks learn shallow learnable examples first?", ICML 2019 workshop on identifying and understanding deep learning phenomena

Difficulty beyond (curriculum) learning

Assessing difficulty is interesting beyond curriculum learning

Example: estimating the difficulty with respect to annotation cost



(a) Labeled (and partially labeled) examples to build models

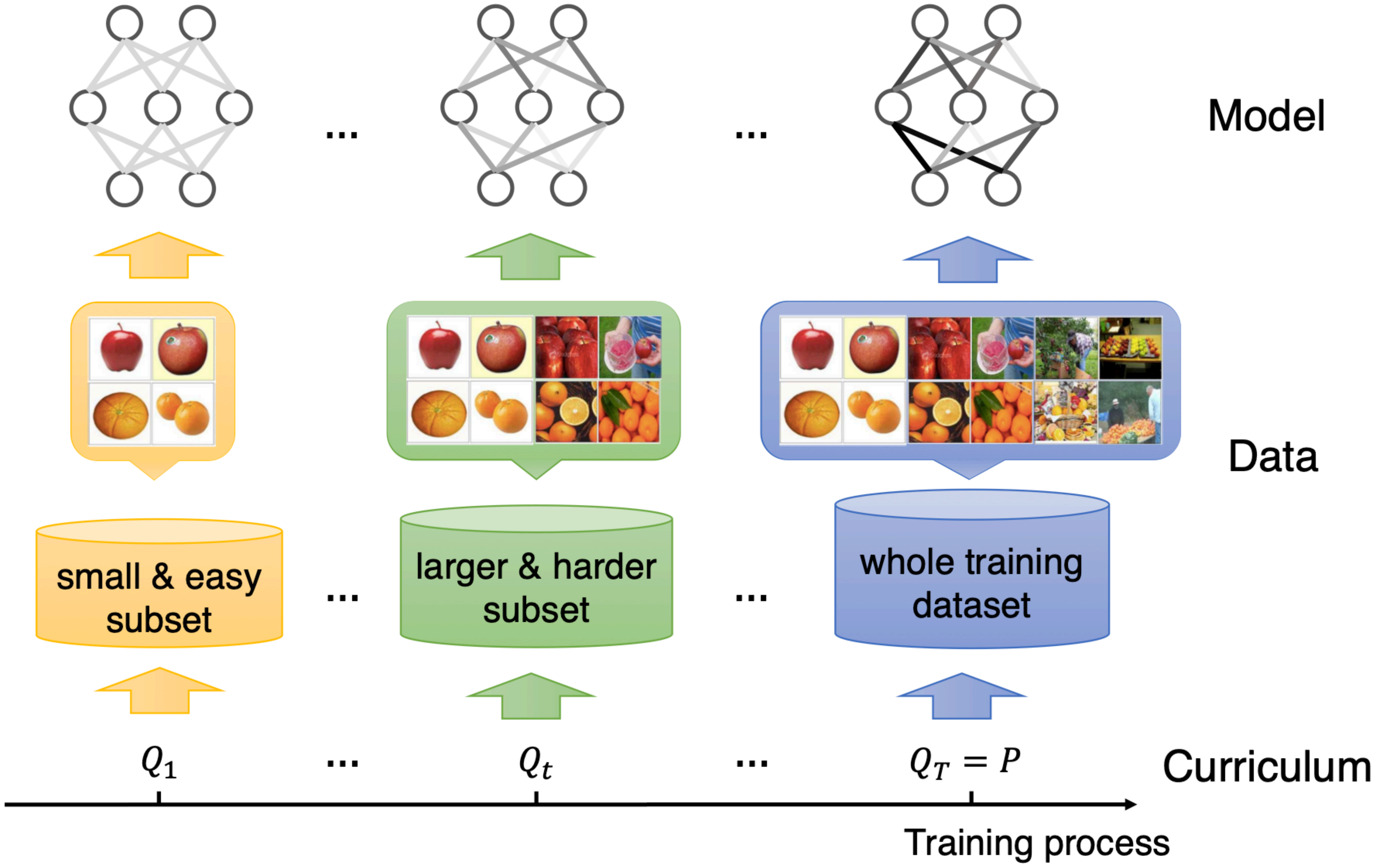
(b) Unlabeled and partially labeled examples to survey

(c) Actively chosen queries sent to annotators

Vijayanarasimhan & Grauman, “What’s It Going to Cost You?: Predicting Effort vs. Informativeness for Multi-Label Image Annotations”, CVPR 2009

Pacing: how to schedule the training

Scheduling training



If we want to define the curriculum up-front, according to prior knowledge, then:

when do we introduce more difficult examples?

Wang et al, “A Survey on Curriculum Learning”, TPAMI 2021

Pacing functions

Various options & heuristics are conceivable

Algorithm 1 One-Pass Curriculum

procedure OP-CURRICULUM(M,$\mathcal{D}$, $\mathcal{C}$)

  $\mathcal{D}' = \text{sort}(\mathcal{D}, \mathcal{C})$

  $\{\mathcal{D}^1, \mathcal{D}^2, ..., \mathcal{D}^k\} = \mathcal{D}'$ where $\mathcal{C}(d_a) < \mathcal{C}(d_b)$ $d_a \in D^i$, $d_b \in D^j, \forall i < j$

  for $s = 1...k$ **do**

    while not converged for p epochs **do**

      train(M, $\mathcal{D}^s$)

    end while

  end for

end procedure

Algorithm from Cirik et al, “Visualizing and understanding curriculum learning for long short-term memory networks”, arXiv, 2016
Based on the procedure described in Bengio et al, “Curriculum Learning”, ICML 2009

Pacing functions

Various options & heuristics are conceivable

Algorithm 1 One-Pass Curriculum

procedure OP-CURRICULUM(M,$\mathcal{D}$, $\mathcal{C}$)
  $\mathcal{D}' = \text{sort}(\mathcal{D}, \mathcal{C})$
  $\{\mathcal{D}^1, \mathcal{D}^2, ..., \mathcal{D}^k\} = \mathcal{D}'$ where $\mathcal{C}(d_a) < \mathcal{C}(d_b)$ $d_a \in D^i$, $d_b \in D^j, \forall i < j$
  for $s = 1...k$ **do**
    while not converged for p epochs **do**
      train(M, $\mathcal{D}^s$)
    end while
  end for
end procedure

Algorithm from Cirik et al, “Visualizing and understanding curriculum learning for long short-term memory networks”, arXiv, 2016
Based on the procedure described in Bengio et al, “Curriculum Learning”, ICML 2009

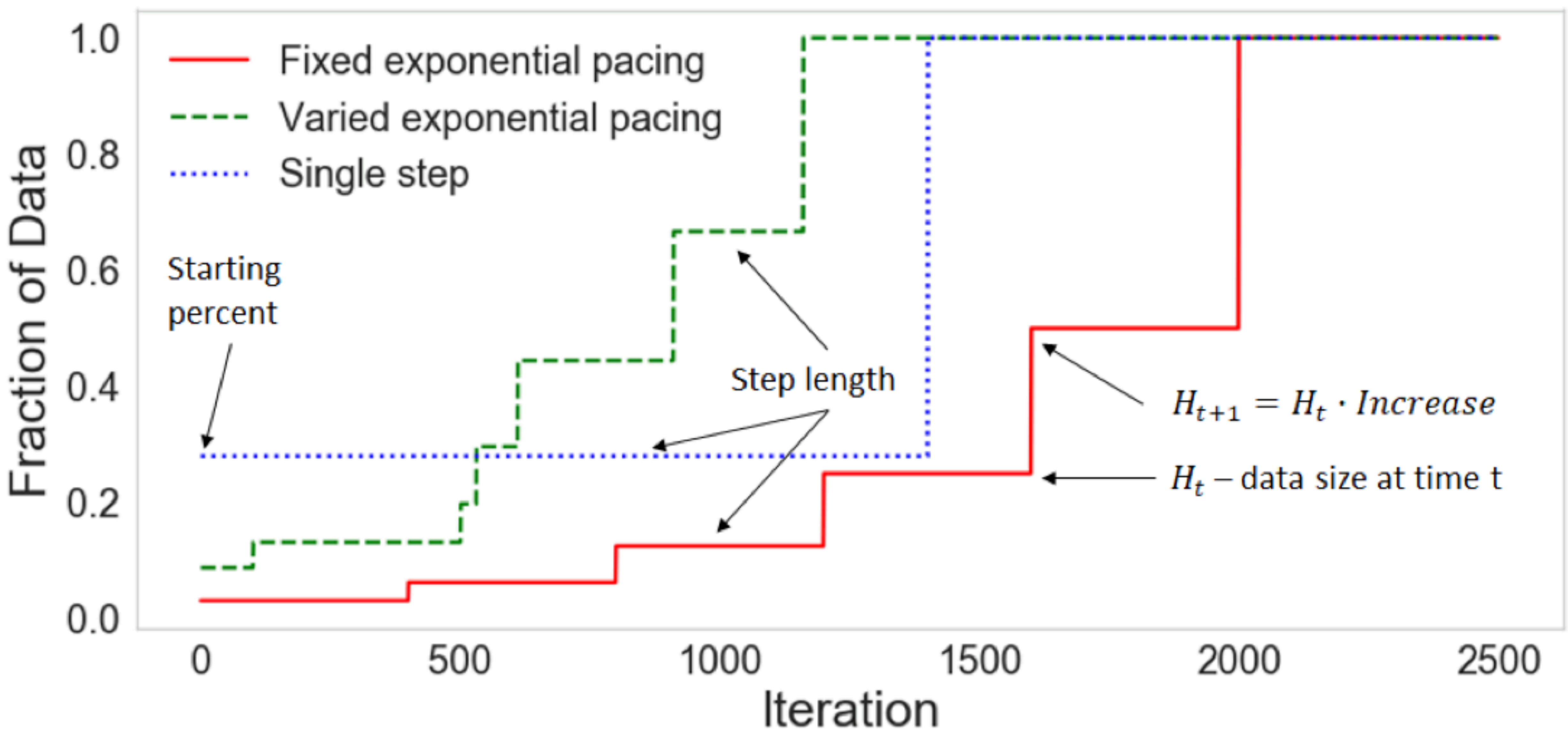
Algorithm 2 Baby Steps Curriculum

procedure BS-CURRICULUM(M,$\mathcal{D}$, $\mathcal{C}$)
  $\mathcal{D}' = \text{sort}(\mathcal{D}, \mathcal{C})$
  $\{\mathcal{D}^1, \mathcal{D}^2, ..., \mathcal{D}^k\} = \mathcal{D}'$ where $\mathcal{C}(d_a) < \mathcal{C}(d_b)$ $d_a \in D^i$, $d_b \in D^j, \forall i < j$
  $\mathcal{D}^{train} = \emptyset$
  for $s = 1...k$ **do**
    $\mathcal{D}^{train} = \mathcal{D}^{train} \cup \mathcal{D}^s$
    while not converged for p epochs **do**
      train(M, $\mathcal{D}^{train}$)
    end while
  end for
end procedure

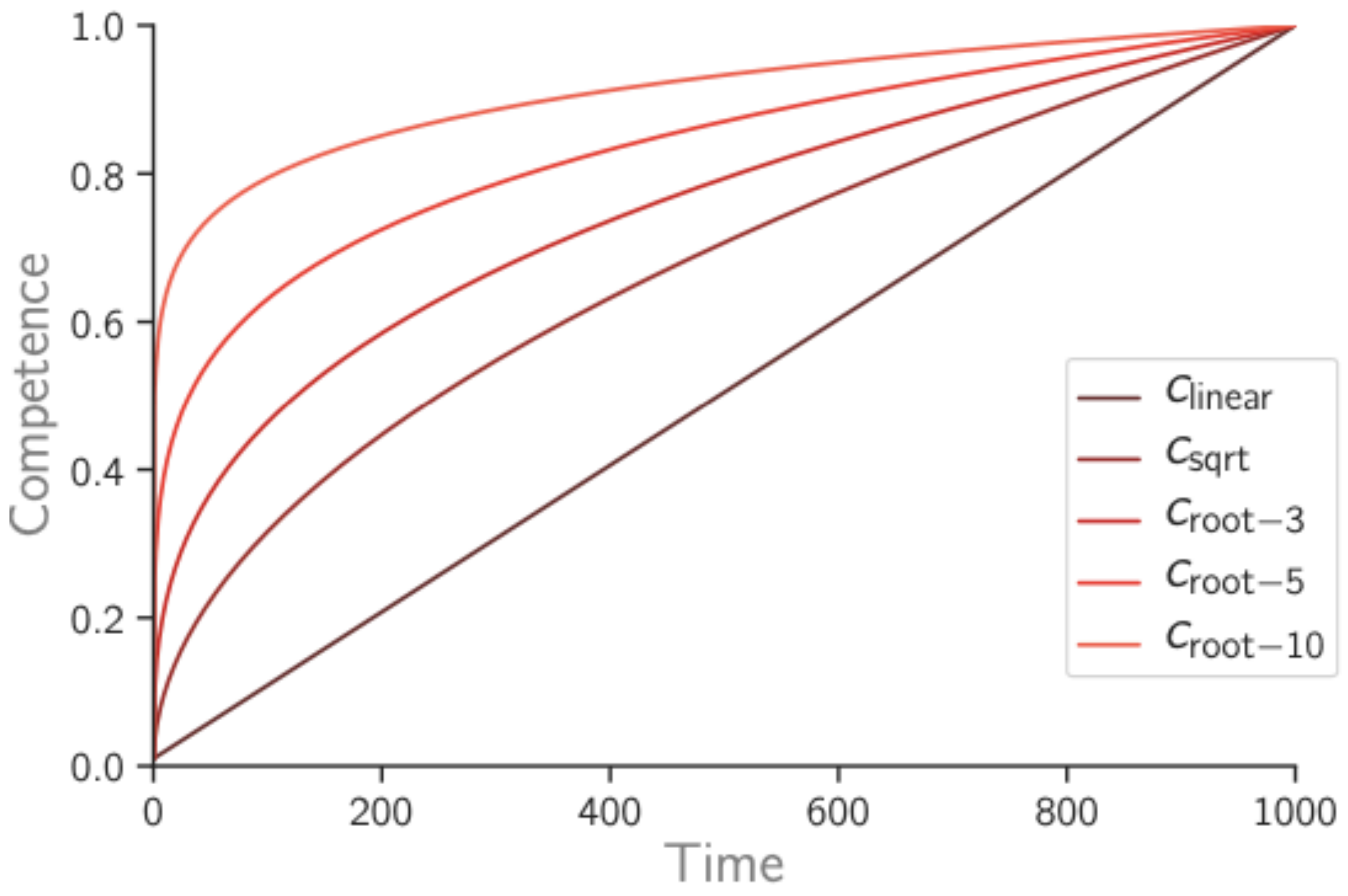
Algorithm from Cirik et al, “Visualizing and understanding curriculum learning for long short-term memory networks”, arXiv, 2016
Based on the procedure described in Spitkovsky et al, “From baby steps to leapfrogs: how less is more in unsupervised dependency parsing”, NAACL-HLT, 2010

Pacing functions

Various options & heuristics are conceivable



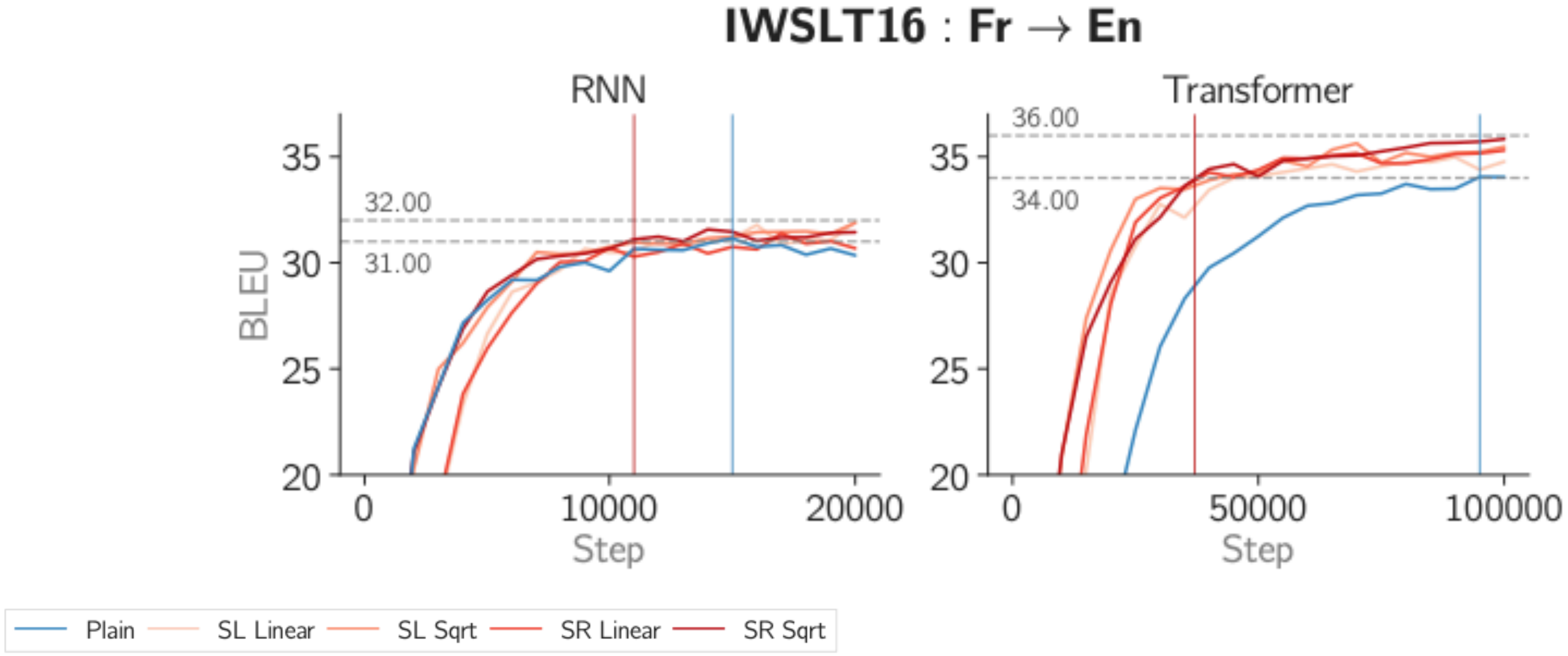
Hacohen & Weinshall, “On the power of curriculum learning in deep networks”, ICML 2019



Platanios et al, “Competence based curriculum learning for neural machine translation”, NAACL-HLT 2019

Pacing functions

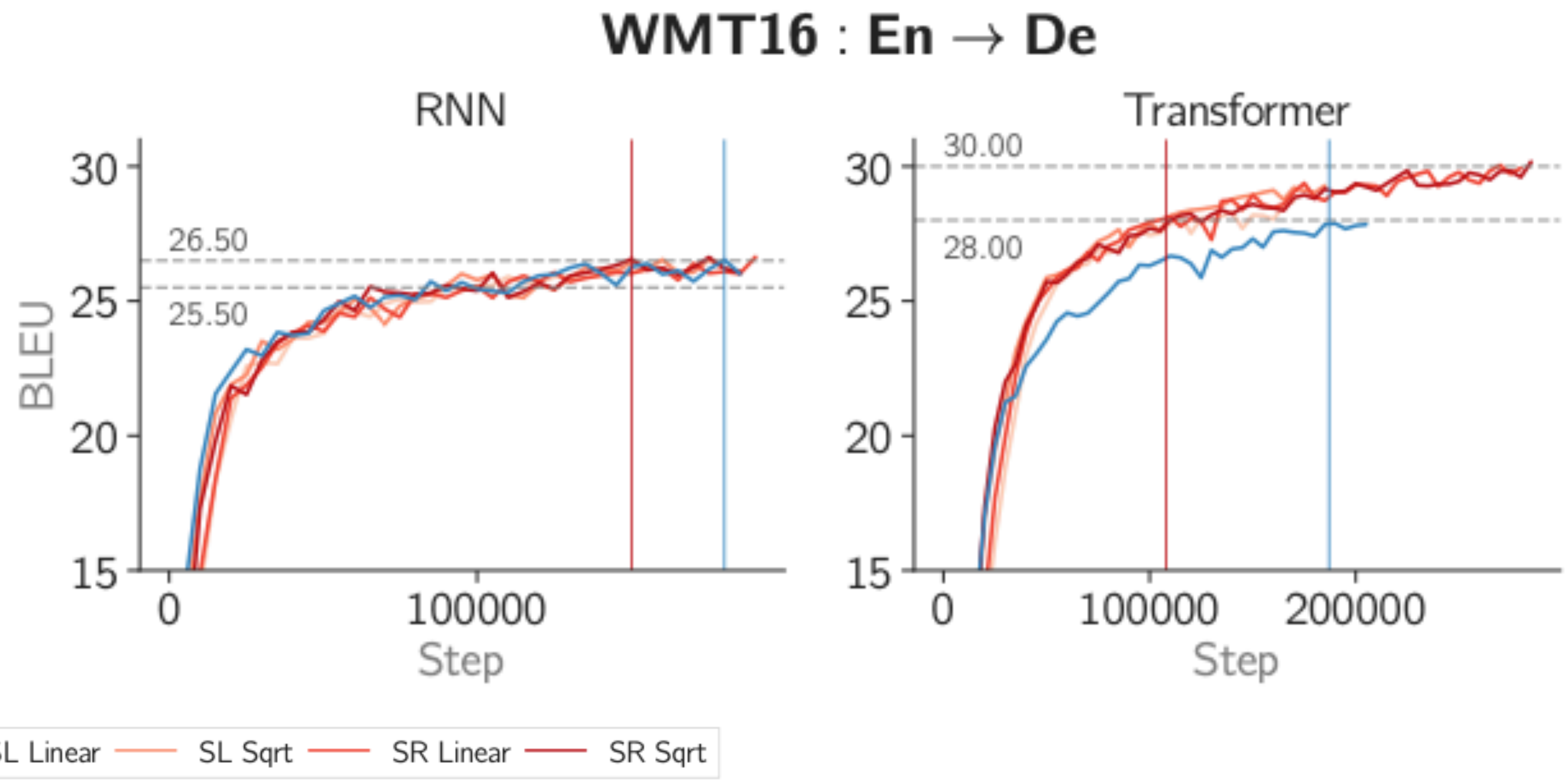
It’s not straightforward to choose, especially due to model/task dependency



Platanios et al, “Competence based curriculum learning for neural machine translation”, NAACL-HIT 2019

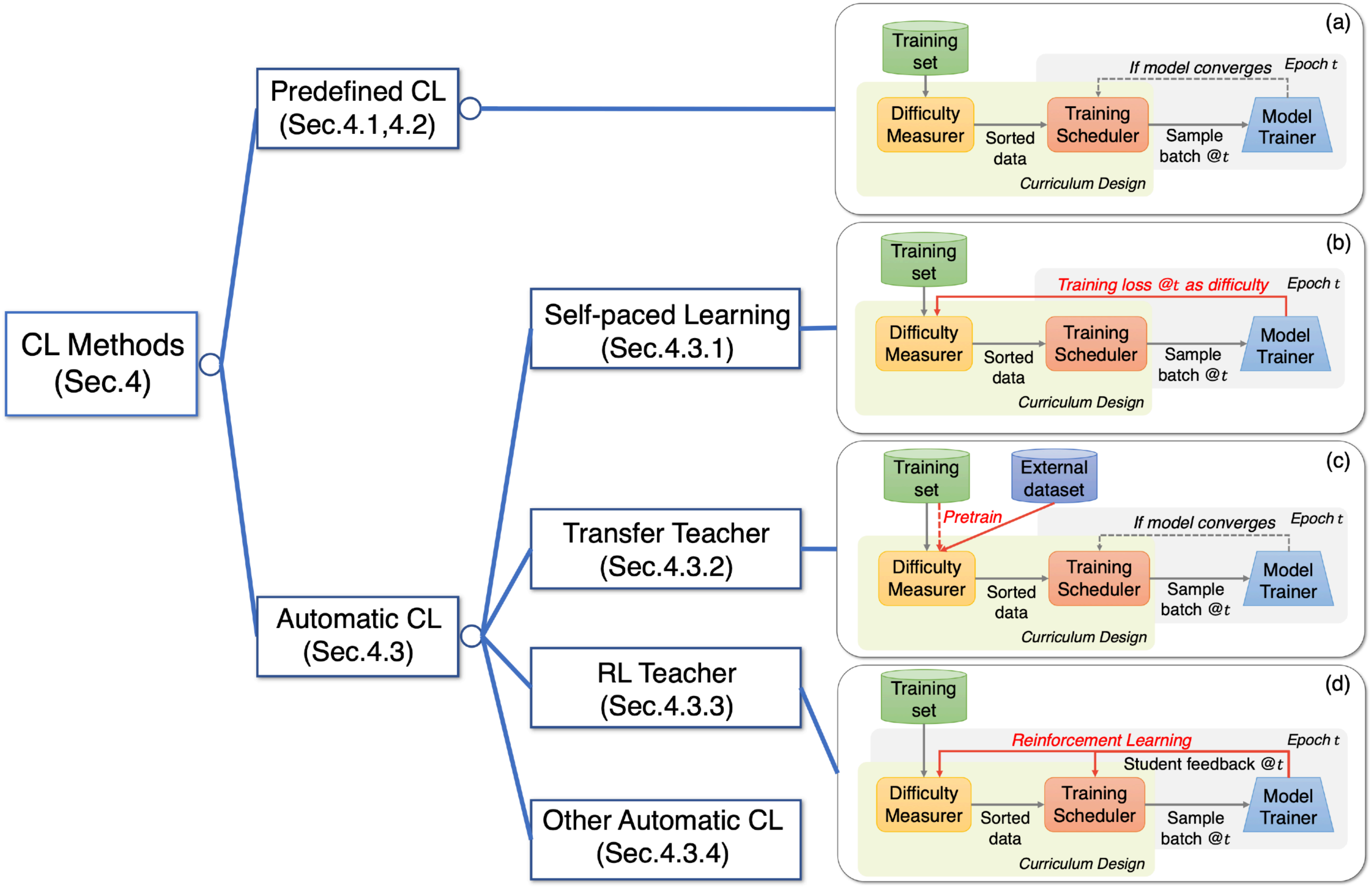
Pacing functions

It’s not straightforward to choose, especially due to model/task dependency



Platanios et al, “Competence based curriculum learning for neural machine translation”, NAACL-HIT 2019

Beyond pre-defined curricula



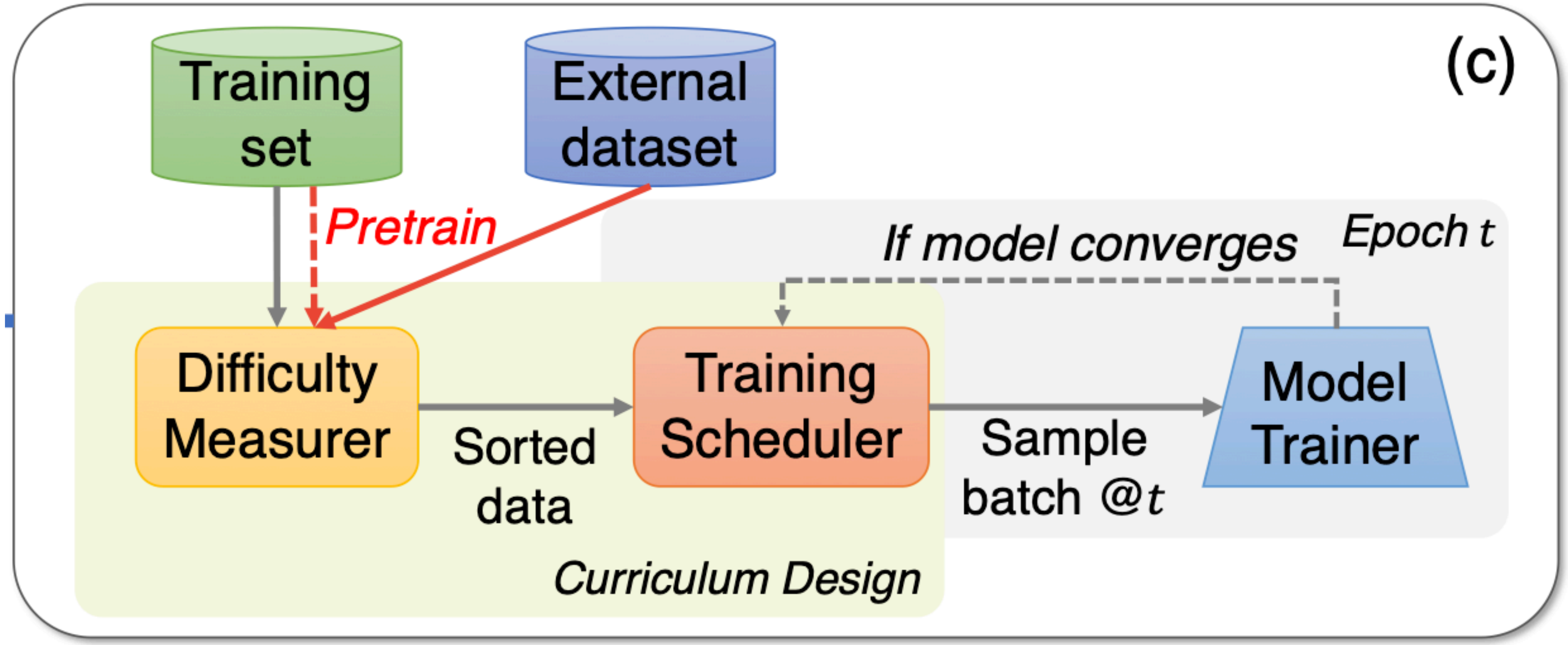
What we just talked about

What we can also consider

Wang et al, “A Survey on Curriculum Learning”, TPAMI 2021

Transfer-teacher curricula

Instead of defining the curriculum, we could use a pre-trained teacher model (based on a different related dataset) based difficulty measure



Wang et al, "A Survey on Curriculum Learning", TPAMI 2021

Transfer-teacher curricula

Instead of defining the curriculum, we could use a pre-trained teacher model (based on a different related dataset) based difficulty measure

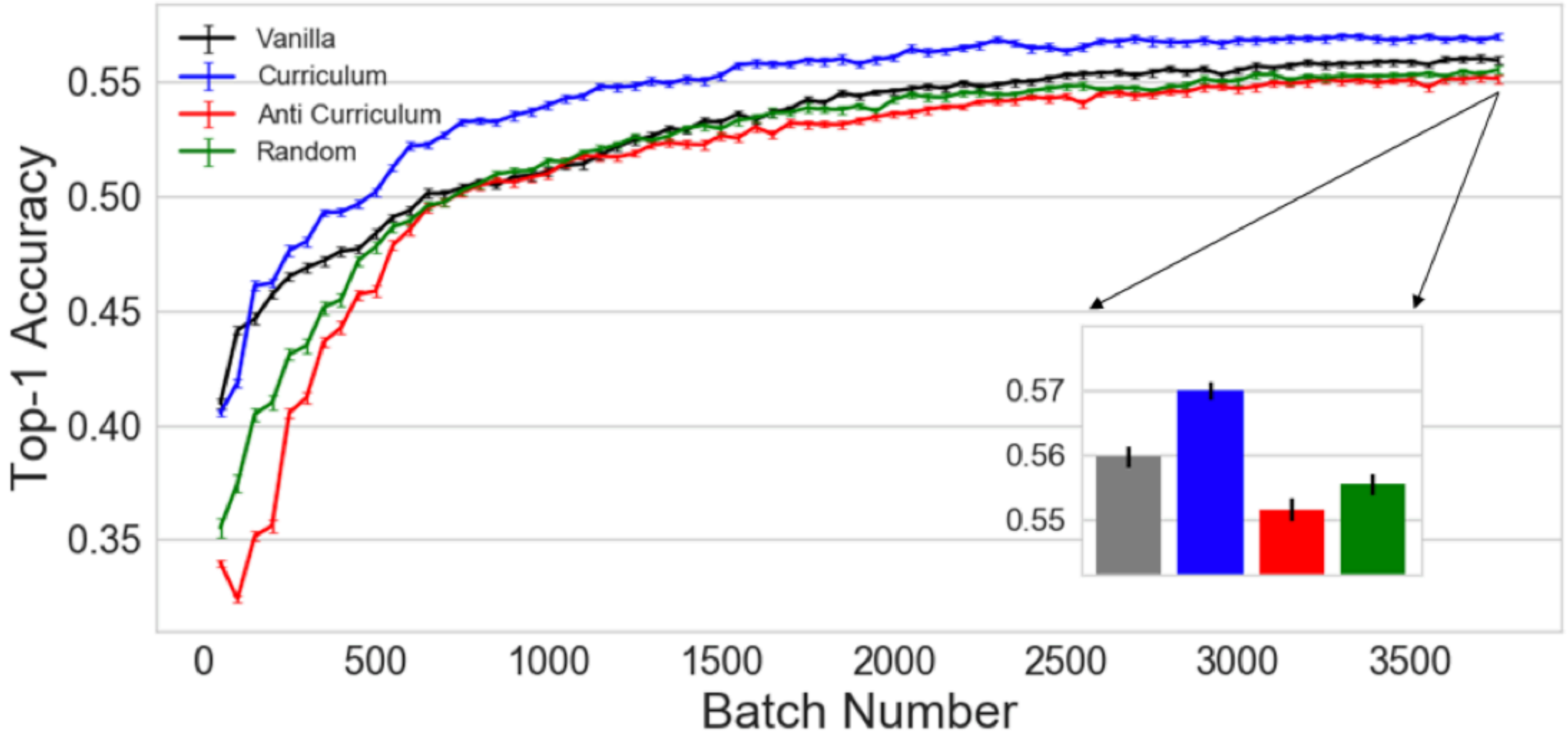


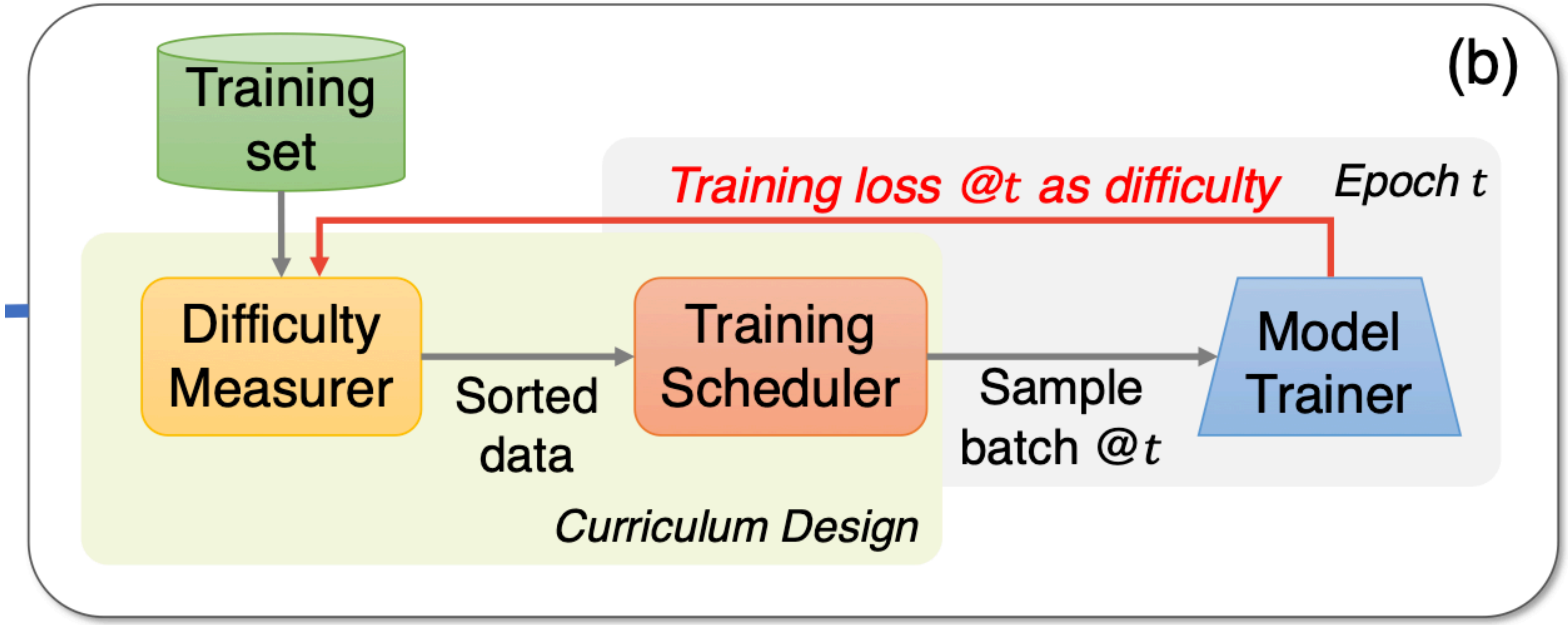
Figure 2. Results in **case 1**, with Inception-based *transfer scoring function* and *fixed exponential pacing function*.

Hacohen & Weinshall, "On the power of curriculum learning in deep networks", ICML 2019

From pre-defined to self-paced

Using a teacher is a form of pre-defined curriculum, what if we want to have an adaptive measure of difficulty, based on our current model?

Moving away from a pre-defined curriculum towards model "competence"

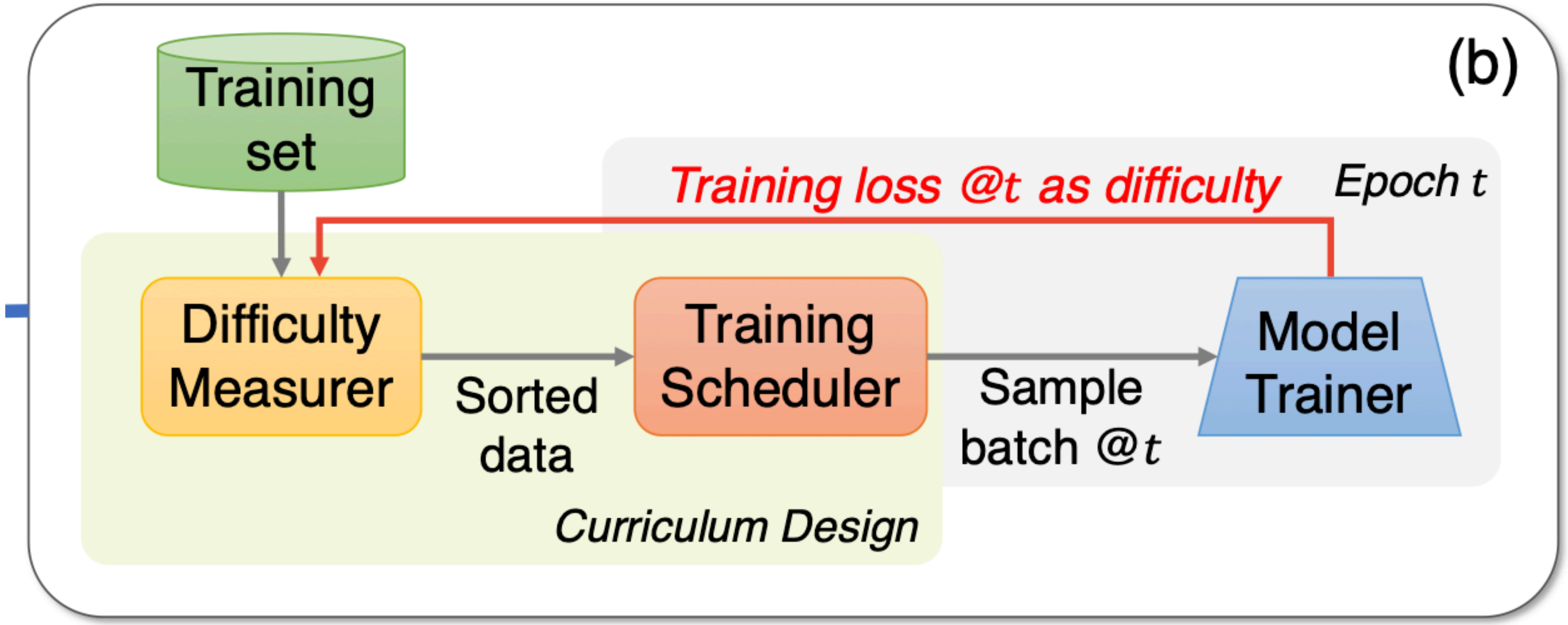


Wang et al, "A Survey on Curriculum Learning", TPAMI 2021

From pre-defined to self-paced

Often this is called self-paced learning

Now rely on a model's current hypothesis at each point in time to assign difficulty to the training data, rather than ranking according to the target



Wang et al, "A Survey on Curriculum Learning", TPAMI 2021

Self-paced & self-taught

Self-paced learning:

Measure the difficulty of an instance according to current loss/predictions etc. (related to the ideas in *active learning*)

Self-taught learning:

Train a model fully, measure each instance according to final model, assign difficulty score and start over with curriculum -> repeat (related to the ideas in *boosting*)

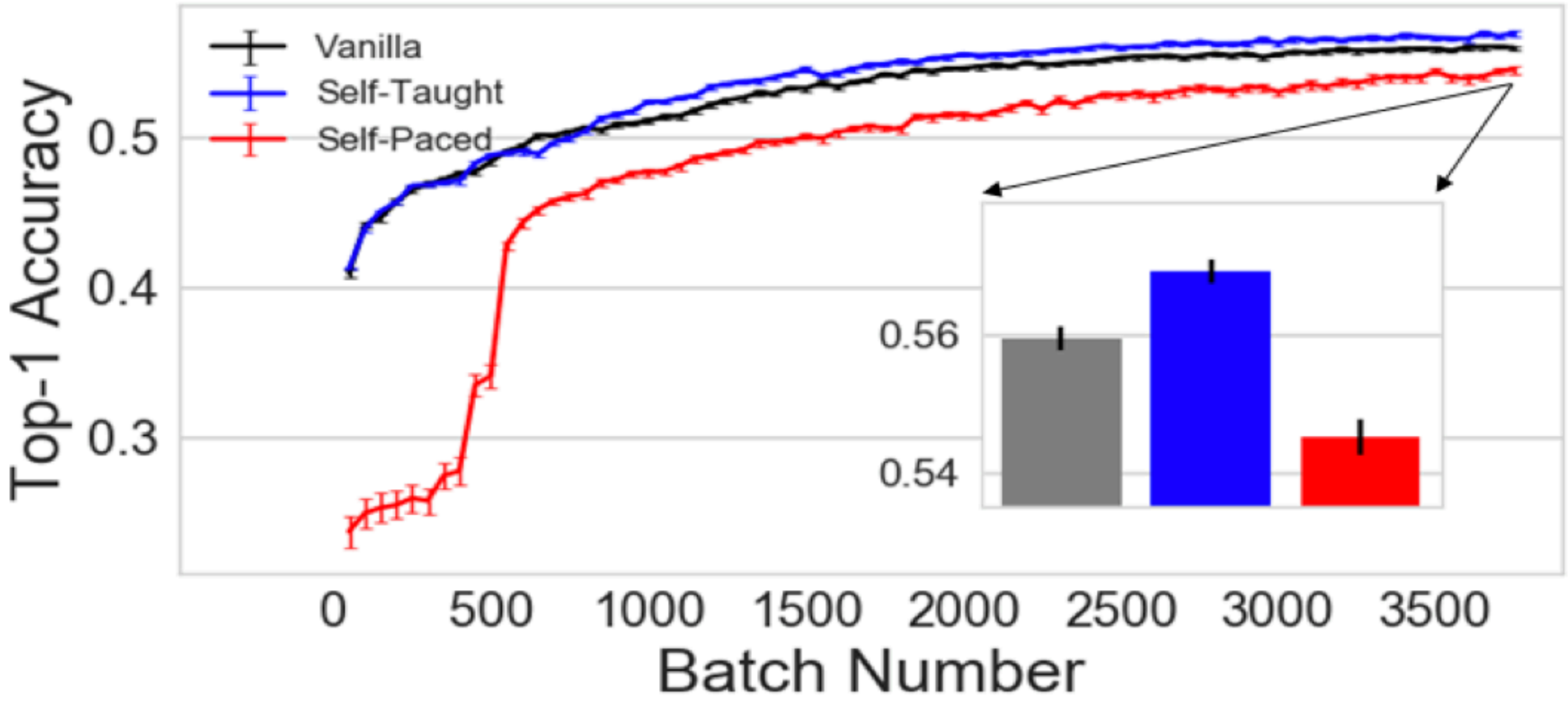
Self-paced & self-taught

Self-paced learning:

Measure the difficulty of an instance according to current loss/predictions etc. (related to the ideas in *active learning*)

Self-taught learning:

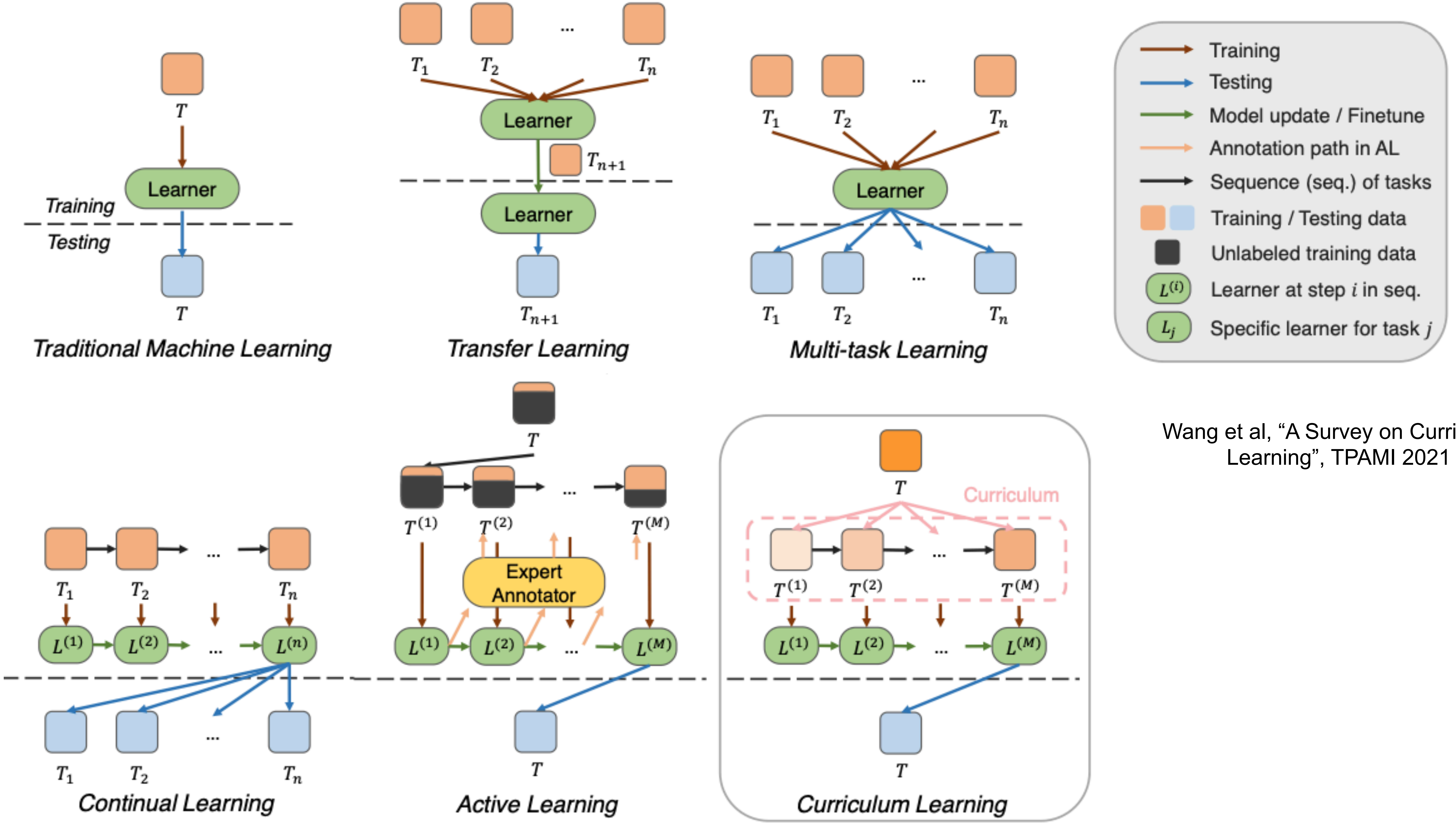
Train a model fully, measure each instance according to final model, assign difficulty score and start over with curriculum -> repeat (related to the ideas in *boosting*)



Hacohen & Weinshall, "On the power of curriculum learning in deep networks", ICML 2019

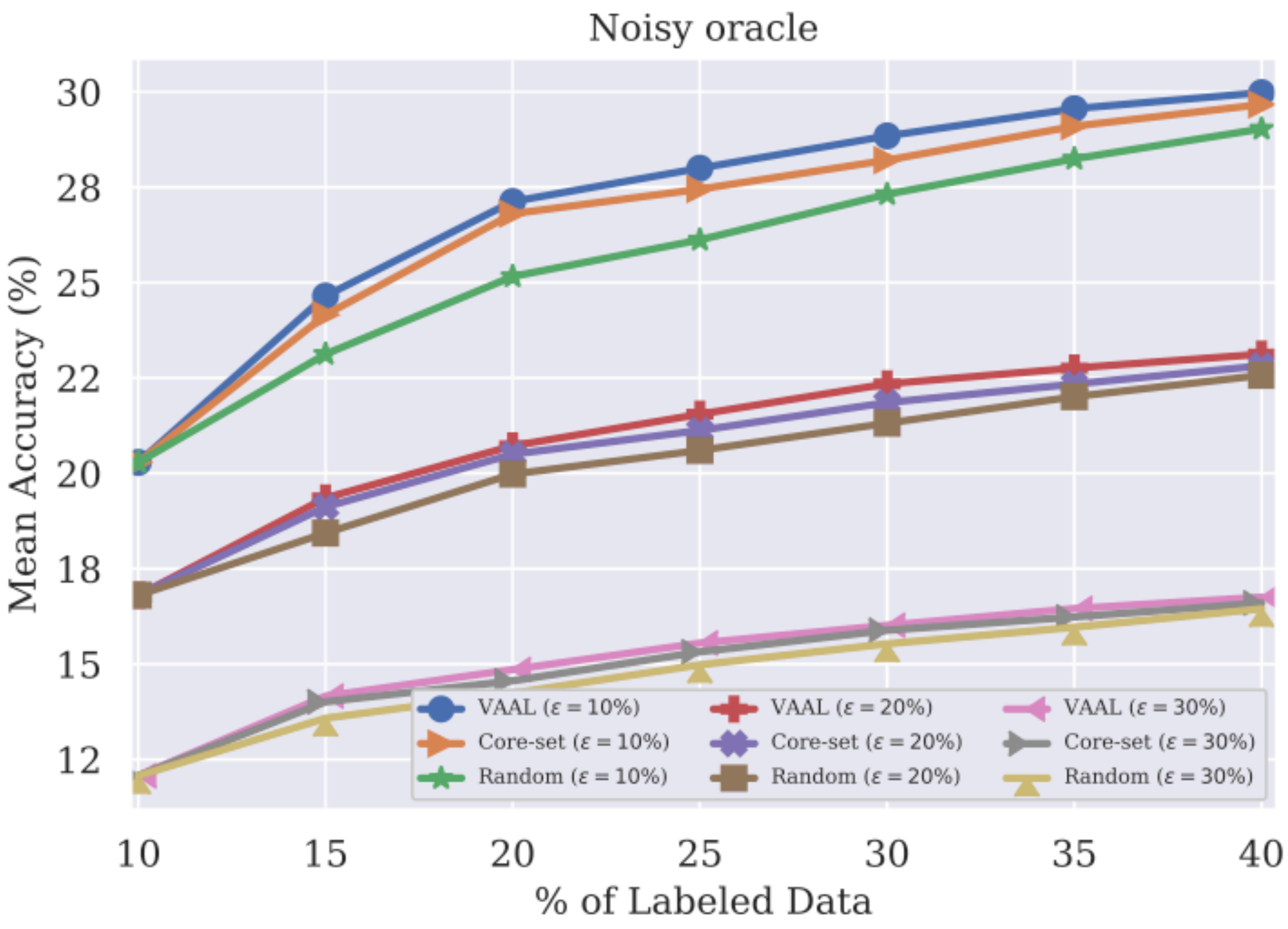
Again: why is it so hard to beat “random”?
"wrong" things to measure & constrained evaluation

It’s about set-up & evaluation (our topic tomorrow)

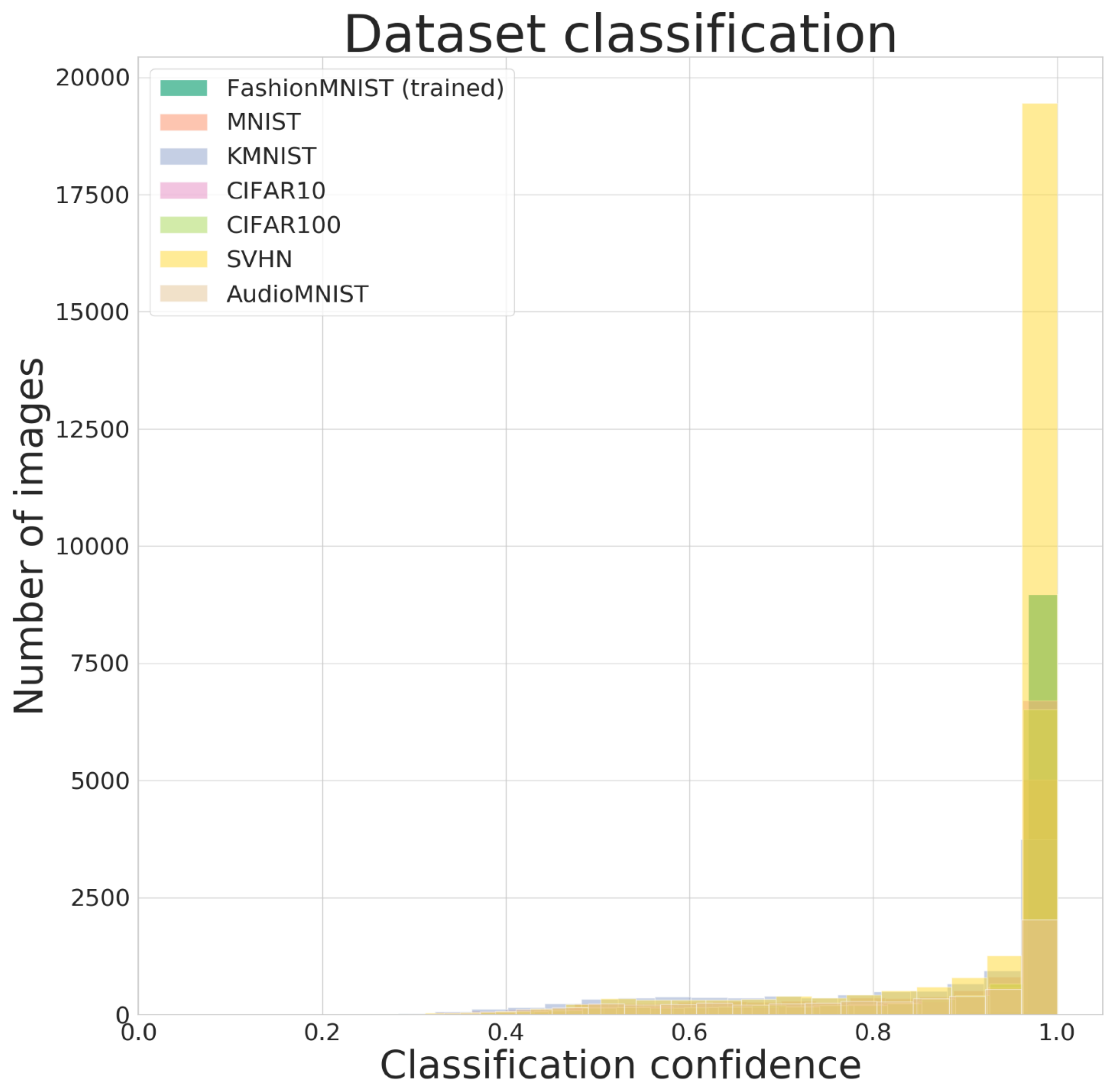


Wang et al, “A Survey on Curriculum Learning”, TPAMI 2021

We have consistently assumed A LOT! Tomorrow's essence: opening "Pandora's box" of evaluation



Sinha et al, "Variational Adversarial Active Learning", ICCV 2019



Mundt et al "Open Set Recognition Through Deep Neural Network Uncertainty, Does Out-of-Distribution Detection Require Generative Classifiers?", ICCV Statistical Deep Learning Workshop 2019 (Based on a long-known problem, Matan1990)