CS388: Natural Language Processing
Lecture 10: Evaluation Principles, Dataset Artifacts

Greg Durrett

TEXAS
The University of Texas at Austin

Announcements

- Final project proposals due next Tuesday

- P3 released next week

Recap

- **Pretraining** (BERT):
  - Train a big model to fill in masked-out words, then adapt it to other tasks. Led to big gains in **question answering** and **NLI** performance. BART/T5, GPT-3, etc. push this further and extend it to other tasks
- **Decoding methods:** nucleus sampling > greedy for open-ended tasks
- **Two tasks we'll focus on today: Question answering** (QA)…
  - "What was Marie Curie the first female recipient of?"
    -> "The Nobel Prize" (find this span in a document)
- …and **NLI**
  - "But I thought you'd sworn off coffee."
    contradicts "I thought that you vowed to drink more coffee."

Today

- Evaluation in NLP: benchmarks and generalization
- Spurious correlations / dataset artifacts
- Debiasing

Cross-Dataset Evaluation

Principles of Evaluation Suites

- Training and testing on i.i.d. data with big neural models often yields very high performance
- “Solving” a task (getting human-level performance) may be useful, but often can’t tell us about our models more broadly
- A parable of single-task evaluation: SWAG

SWAG: Weaknesses of single tasks

On stage, a woman takes a seat at the piano. She
a) sits on a bench as her sister plays with the doll.
b) smiles with someone as the music plays.
c) is in the crowd, watching the dancers.
d) nervously sets her fingers on the keys.

Rowan Zellers et al., 2019

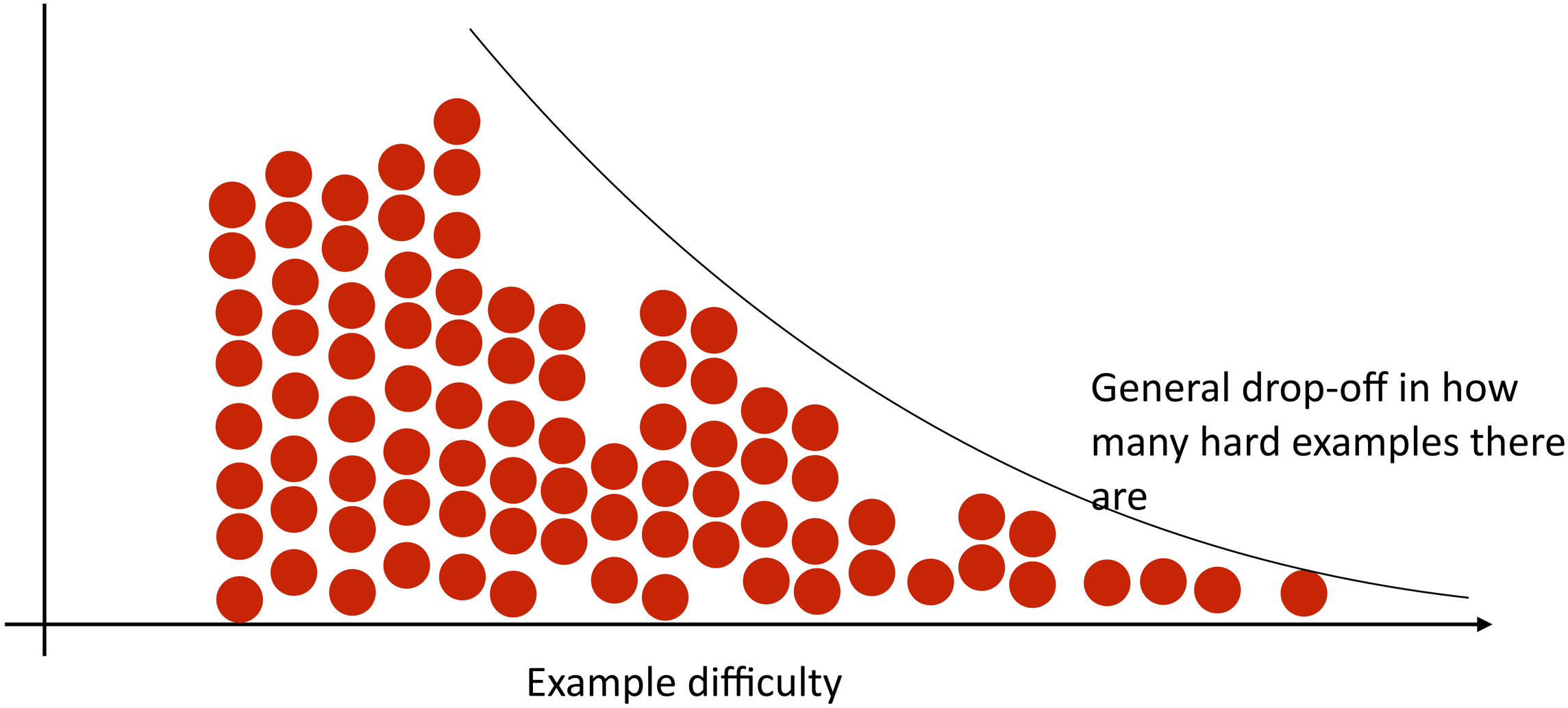
- Text-only data comes from video captions in ActivityNet
- **Adversarial filtering** to produce the negative multiple choice answers
- They said models get ~60% and humans get ~85%, but BERT immediately solved this dataset when it was released

while convergence not reached **do**
- Split the dataset $\mathcal{D}$ randomly up into training and testing portions $\mathcal{D}^{tr}$ and $\mathcal{D}^{te}$.
- Optimize a model f_θ on $\mathcal{D}^{tr}$.

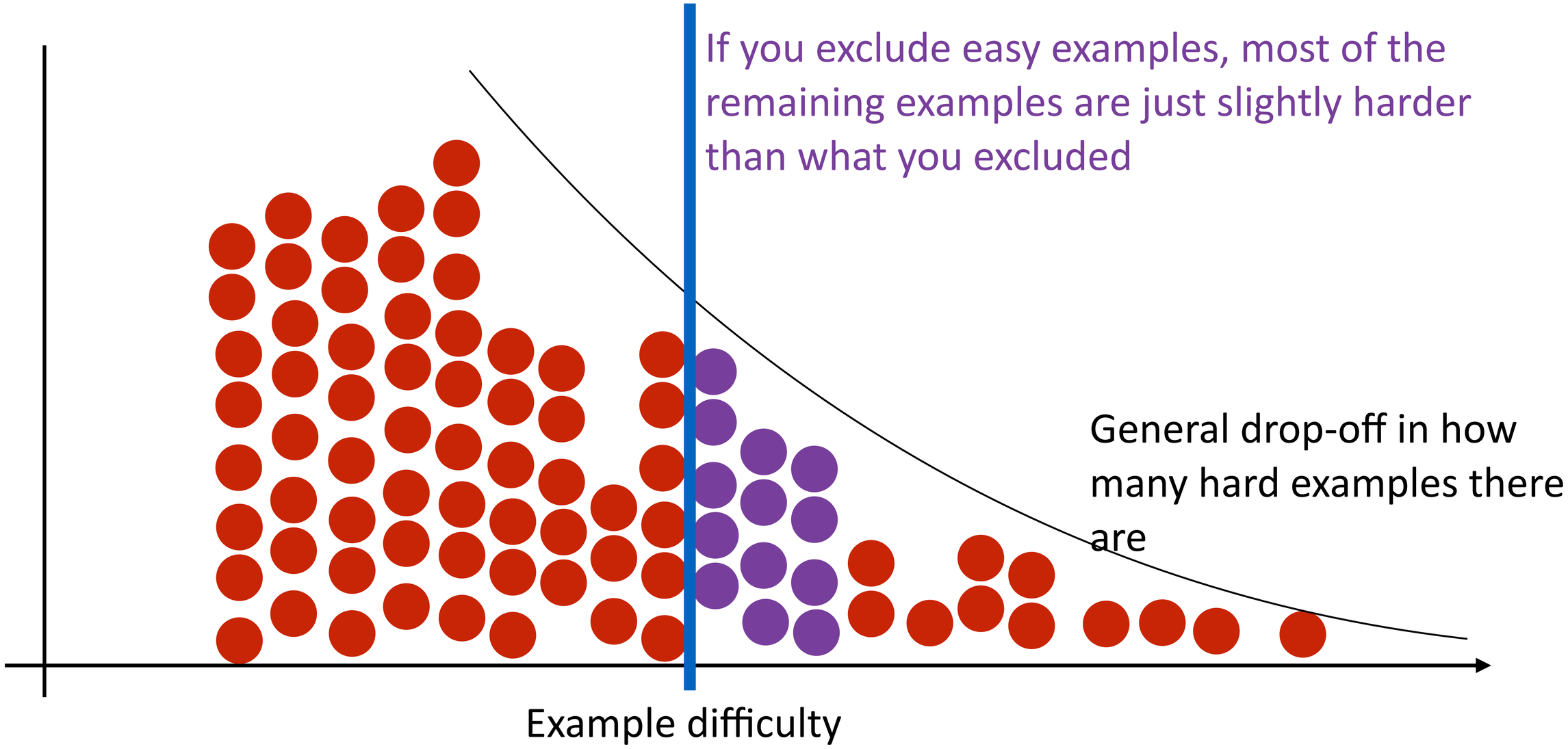
for index i in $\mathcal{D}^{te}$ **do**
- Identify easy indices: $\mathcal{A}_i^{easy} = \{j \in \mathcal{A}_i : f_\theta(x_i^+) > f_\theta(x_{i,j}^-)\}$
- Replace N^{easy} easy indices $j \in \mathcal{A}_i^{easy}$ with adversarial indices $k \notin \mathcal{A}_i$ satisfying $f_\theta(x_{i,k}^-) > f_\theta(x_{i,j}^-)$.

end for
end while

Intuition


General drop-off in how many hard examples there are
Example difficulty

Intuition


If you exclude easy examples, most of the remaining examples are just slightly harder than what you excluded
General drop-off in how many hard examples there are
Example difficulty

Principles of Evaluation Suites

- Training and testing on i.i.d. data with big neural models often yields very high performance
- "Solving" a task (getting human-level performance) may be useful, but often can't tell us about our models more broadly
- **Designing a single, difficult task is really challenging!**
- To assess big models, we need **evaluation suites (benchmarks)** like GLUE
- What makes a good evaluation suite of tasks?

Principles of Evaluation Suites

- Difficulty: even if some task can be solved by hand-engineering, it should be hard to solve all *N* tasks
- Diverse: doing well on it should say something useful
- Good “yardstick”: should understand where human performance is and what good performance on the task would mean
- GLUE was the first of these, but it wasn’t really diverse and it was too easy. Next step: SuperGLUE

Alex Wang et al., 2019

SuperGLUE: Task Requirements

- Task substance: "*Tasks should test a system's ability to understand and reason about texts in English.*"
- Task difficulty: "*Tasks should be beyond the scope of current state-of-the-art systems, but solvable by most college-educated English speakers.*" (notably they excluded domain-specific tasks, which have become more popular these days, e.g., the bar exam)
- Evaluatable: this is challenging to find!
- Public dataset, good license, etc.

Alex Wang et al., 2019

SuperGLUE: Performance

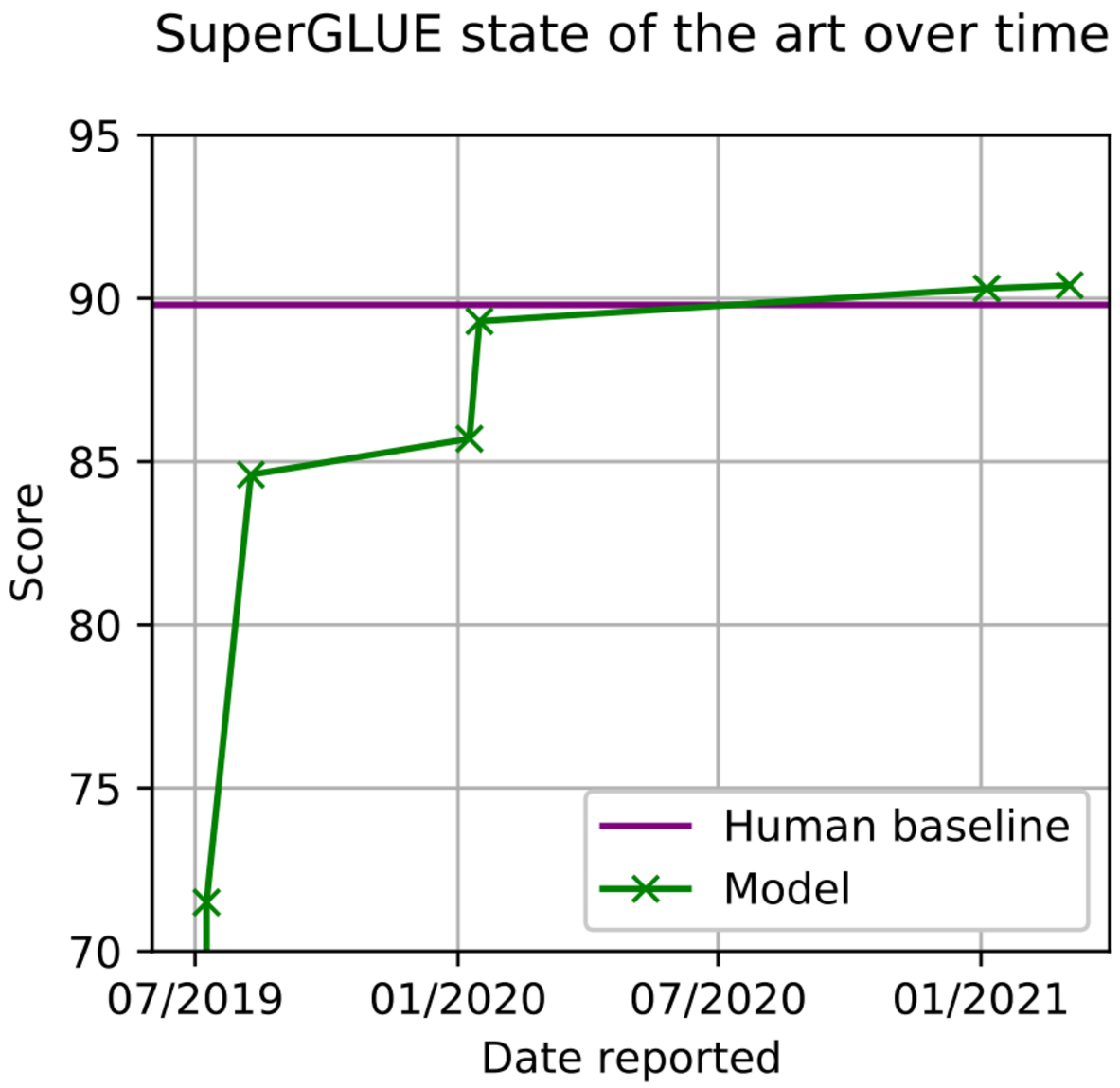
Model Metrics	Avg	BoolQ Acc.	CB F1/Acc.	COPA Acc.	MultiRC $F1_a$/EM	ReCoRD F1/EM	RTE Acc.	WiC Acc.	WSC Acc.	AX_b MCC	AX_g GPS	Acc.
Most Frequent	47.1	62.3	21.7/48.4	50.0	61.1 / 0.3	33.4/32.5	50.3	50.0	65.1	0.0	100.0/	50.0
CBoW	44.3	62.1	49.0/71.2	51.6	0.0 / 0.4	14.0/13.6	49.7	53.0	65.1	-0.4	100.0/	50.0
BERT	69.0	77.4	75.7/83.6	70.6	70.0 / 24.0	72.0/71.3	71.6	**69.5**	**64.3**	23.0	97.8 /	51.7
BERT++	**71.5**	79.0	**84.7/90.4**	73.8	70.0 / 24.1	72.0/71.3	79.0	**69.5**	**64.3**	38.0	99.4 /	51.4
Outside Best	-	**80.4**	- / -	**84.4**	**70.4*/24.5***	**74.8/73.0**	**82.7**	-	-	-	- /	-
Human (est.)	89.8	89.0	95.8/98.9	100.0	81.8*/51.9*	91.7/91.3	93.6	80.0	100.0	77.0	99.3 /	99.7

- RoBERTa in 2019: 84.6
- DeBERTa in 2020: 90.3. **Even SuperGLUE was solved quickly!**

Alex Wang et al., 2019

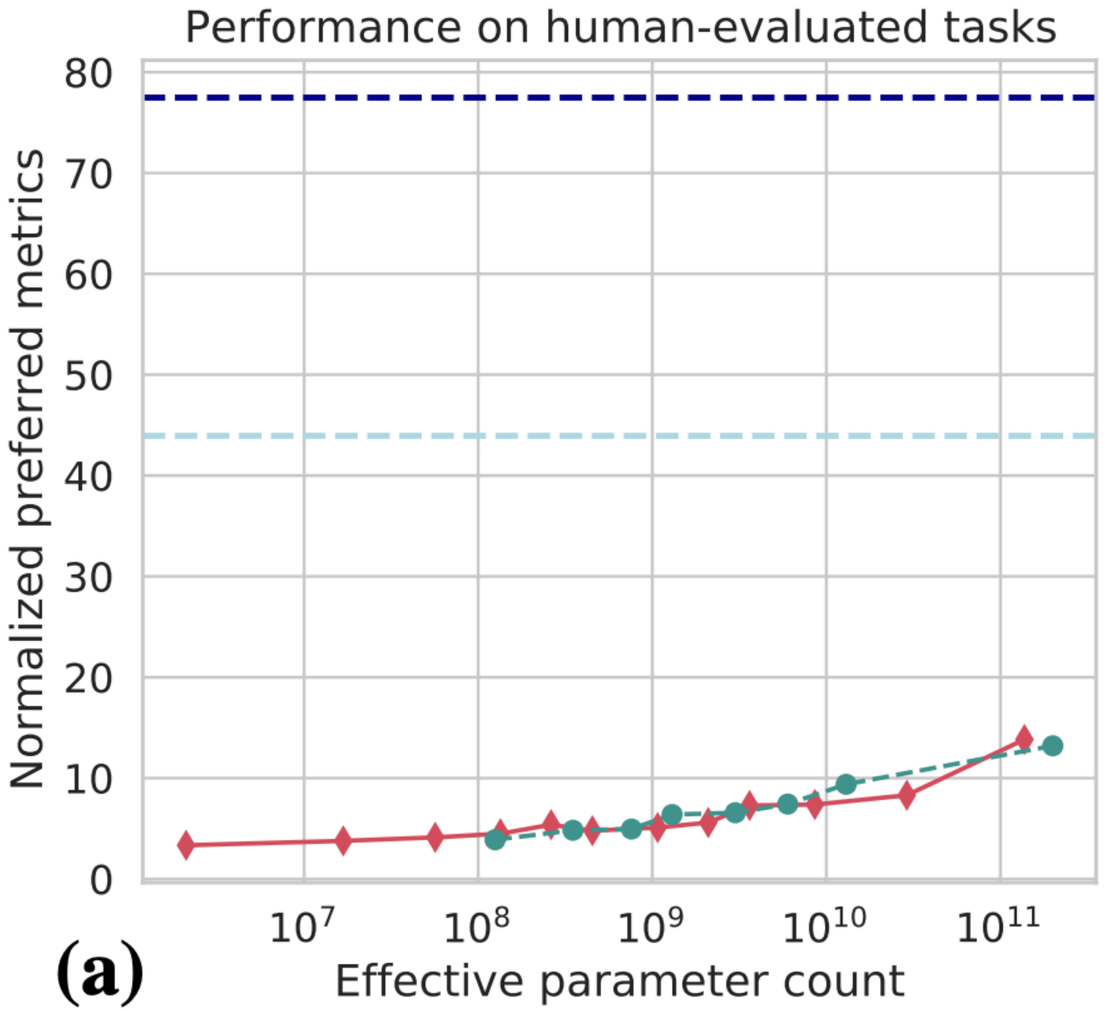
SuperGLUE: Performance

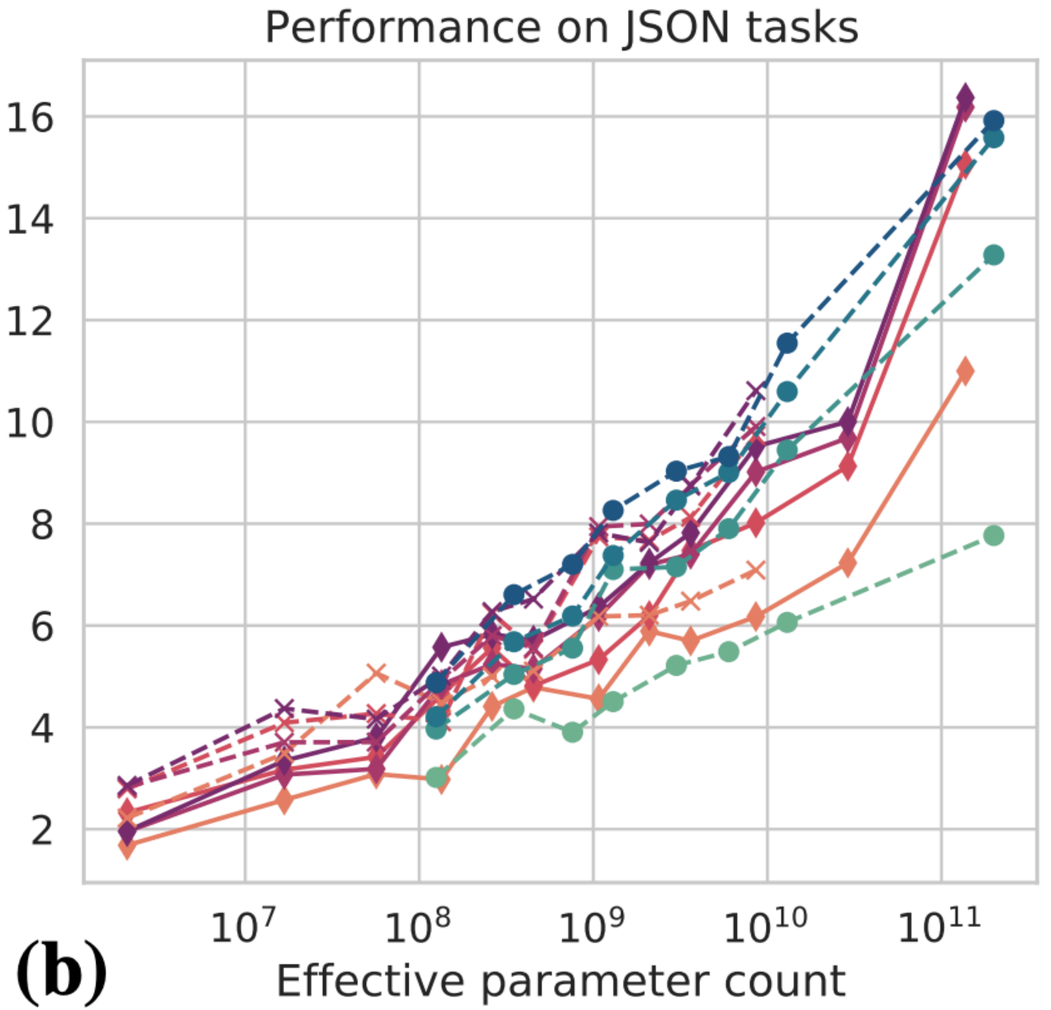
As reported in BIGBench:



BIG-bench

- 204 tasks, 444 authors





BIG-bench

- "Beyond the Imitation Game" — aim to learn more than what's possible from model vs. human performance

- Particular emphasis on scaling

- Primarily for pre-trained models without fine-tuning. Therefore, not all tasks have large training (or even test!) sets

MMLU

- MMLU task (Hendrycks et al., 2020): 57 high school/college/professional exams:

Conceptual Physics

When you drop a ball from rest it accelerates downward at 9.8 m/s². If you instead throw it downward assuming no air resistance its acceleration immediately after leaving your hand is

(A) 9.8 m/s² ✔
(B) more than 9.8 m/s² ✘
(C) less than 9.8 m/s² ✘
(D) Cannot say unless the speed of throw is given. ✘

College Mathematics

In the complex z-plane, the set of points satisfying the equation $z^2 = |z|^2$ is a

(A) pair of points ✘
(B) circle ✘
(C) half-line ✘
(D) line ✔

Figure 4: Examples from the Conceptual Physics and College Mathematics STEM tasks.

Chung et al. (2022)

MMLU

- MMLU task (Hendrycks et al., 2020): 57 high school/college/professional exams:

-	Random	25.0
-	Average human rater	34.5
May 2020	GPT-3 5-shot	43.9
Mar. 2022	Chinchilla 5-shot	67.6
Apr. 2022	PaLM 5-shot	69.3
Oct. 2022	**Flan-PaLM 5-shot**	**72.2**
	Flan-PaLM 5-shot: CoT + SC	**75.2**
-	Average human expert	89.8

Chung et al. (2022)

MMLU

Model	Finetuning Mixtures	Tasks	Norm. avg.	MMLU		BBH	
				Direct	CoT	Direct	CoT
540B	None (no finetuning)	0	49.1	71.3	62.9	49.1	63.7
	CoT	9	52.6 (+3.5)	68.8	64.8	50.5	61.1
	CoT, Muffin	89	57.0 (+7.9)	71.8	66.7	56.7	64.0
	CoT, Muffin, T0-SF	282	57.5 (+8.4)	72.9	**68.2**	57.3	64.0
	CoT, Muffin, T0-SF, NIV2	1,836	**58.5** (+9.4)	**73.2**	68.1	**58.8**	**65.6**

- Human performance estimates are ~90 on MMLU, ~80 on Big-Bench (BBH). Even getting close on **these** tasks!

Chung et al. (2022)

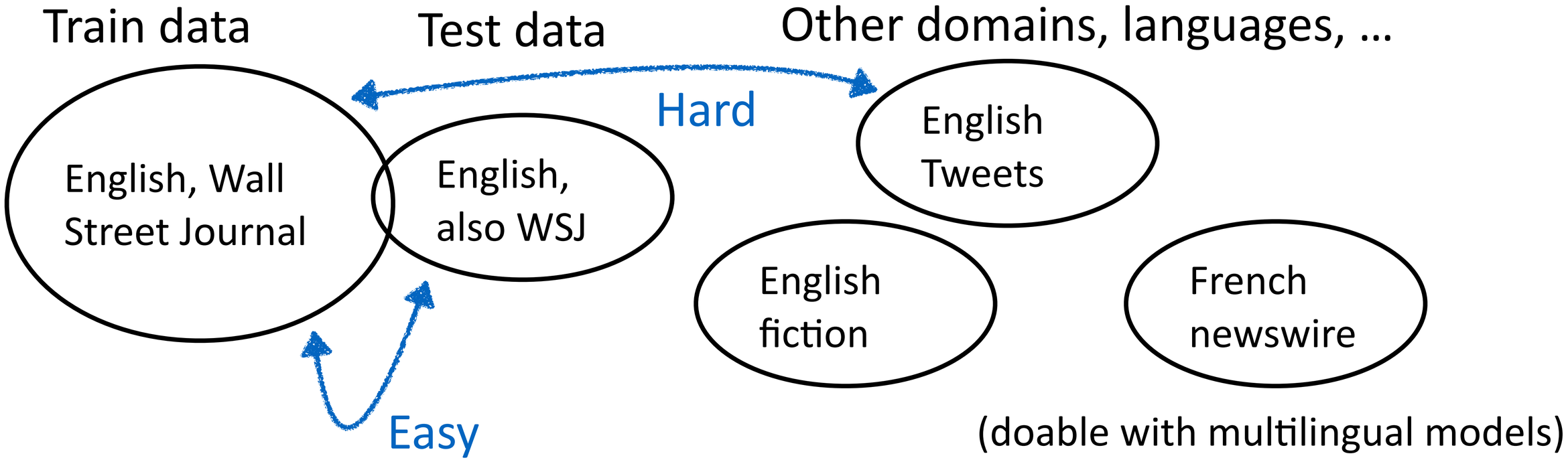
Evaluation Under Distribution Shift

Model Performance

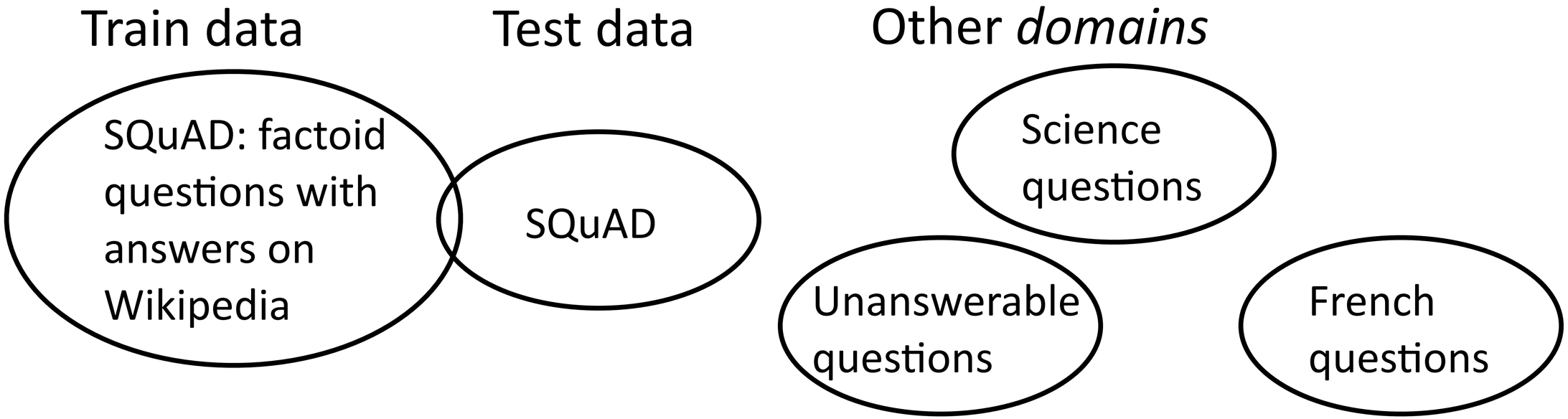
- If models can be fine-tuned on each of n tasks in an evaluation suite and perform very well on the held-out test dataset, have we solved everything we want?
- What can go wrong?

Generalization

- If a model does well on train but poorly on test data, it *doesn't generalize*
- A model can do well on its test data and still fail to generalize *out of distribution* — arguably an even more important notion
- Many notions of generalization. Example: POS tagging



Generalization: QA



Other types of reasoning, such as *multi-hop questions*

Who won the Nobel in Chemistry the year Marie Curie won the Nobel in Physics?

Generalization

- Just doing well on a single test set **is not that useful**
- We want POS taggers, QA systems, and more that can generalize to new settings so we can deploy them in practice
- Sometimes, you can get **very good test performance** but the model **generalizes very poorly**. How does this happen?

Annotation Artifacts, Reasoning Shortcuts: QA

Annotation Artifacts

- Some datasets might be easy because of how they're constructed, especially in QA and NLI

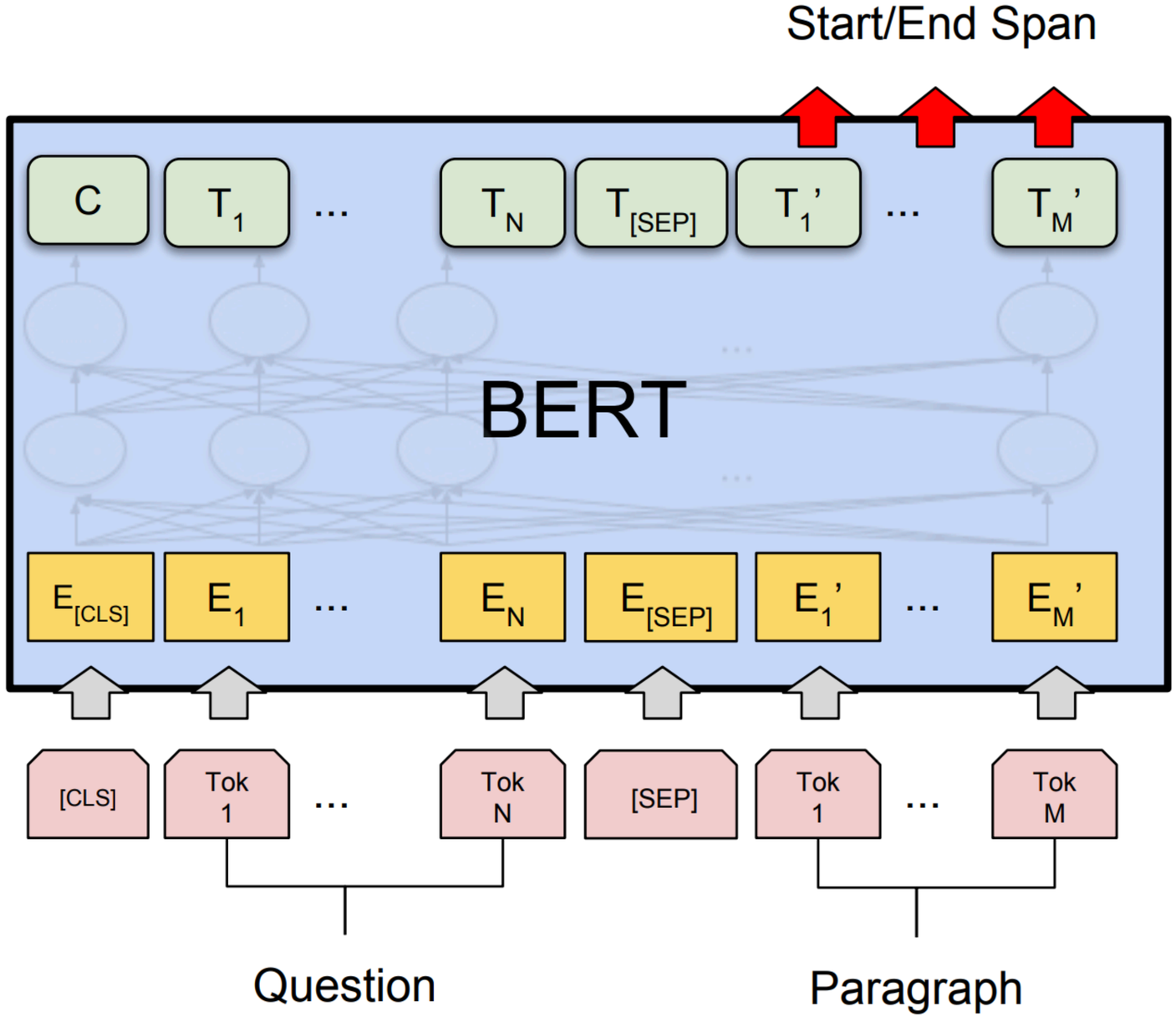
  What becomes of Macbeth?

  What does Macduff do to Macbeth?

  What violent act does Macduff perform upon Macbeth?

- All questions have the same answer. But some are more easily guessable

Reminder: QA with BERT



Devlin et al. (2019)

QA: Answer Type Heuristics

What degree did Martin Luther receive on October 19, 1512?

On October 19, 1512, Luther was awarded his doctorate of theology and, on October 21, 1512, was received into the senate of the theological faculty of the University of Wittenberg. He spent the rest of his career in this position at the University of Wittenberg.

- What should the model be doing? Corresponding Martin Luther with Luther, matching October 19, 1512 between question and passage

QA: Answer Type Heuristics

What degree did Martin Luther receive?

What degree ___?

On October 19, 1512, Luther was awarded his doctorate of theology and, on October 21, 1512, was received into the senate of the theological faculty of the University of Wittenberg. He spent the rest of his career in this position at the University of Wittenberg.

- Only one possible degree here! Model only needs to see “what degree” and will not learn to use the rest of the context!

QA: Answer Type Heuristics

- Question type is powerful indicator. Only a couple of locations in this context!

Where ____?

On October 19, 1512, Luther was awarded his doctorate of theology and, on October 21, 1512, was received into the senate of the theological faculty of the University of Wittenberg. He spent the rest of his career in this position at the University of Wittenberg.

Who ____?

When ____?

QA: Answer Type Heuristics

- Question type is powerful indicator. Only a couple of locations in this context!

  Where ____? *Who ____?* *When ____?*

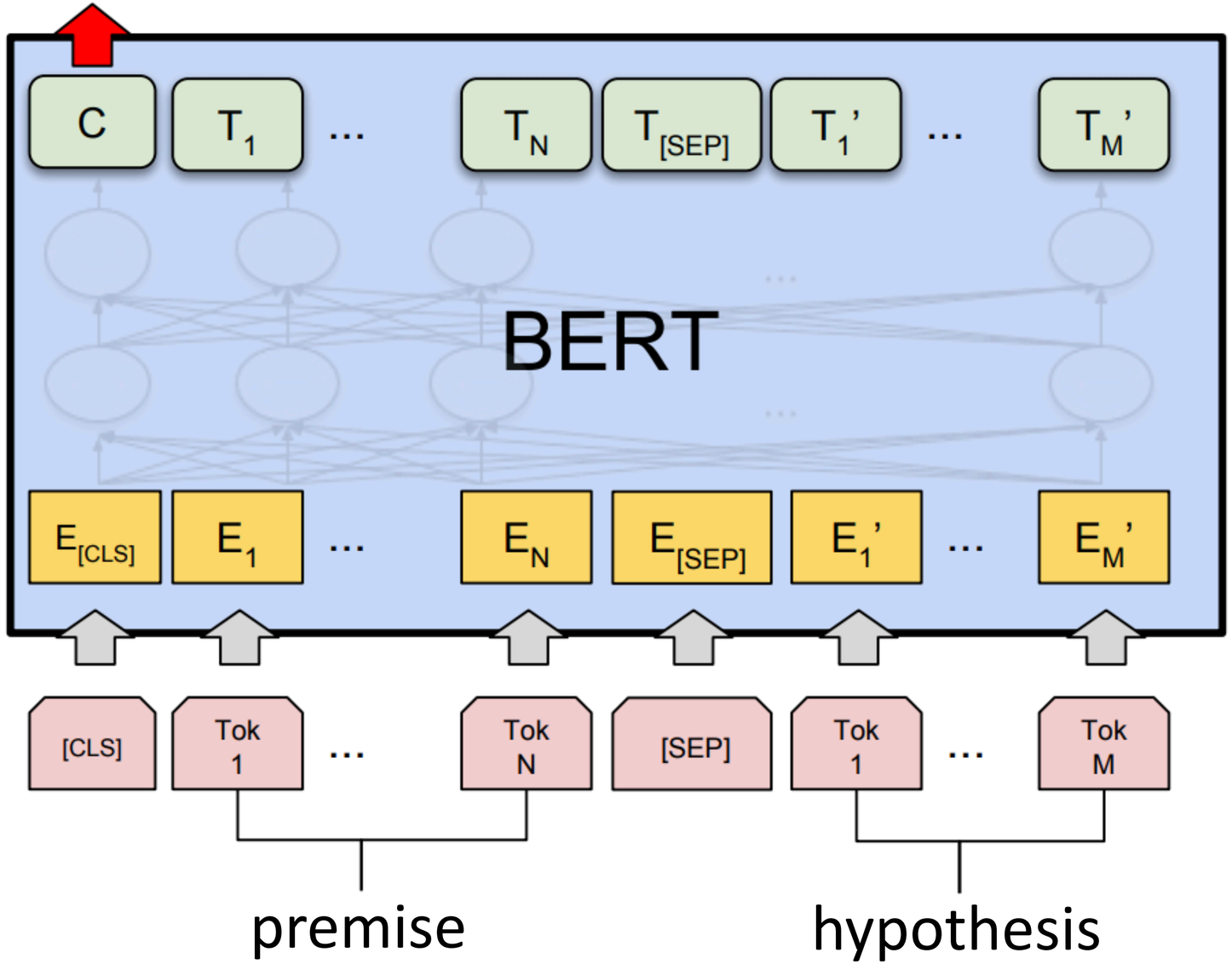
  On October 19, 1512, Luther was awarded his doctorate of theology and, on October 21, 1512, was received into the senate of the theological faculty of the University of Wittenberg. He spent the rest of his career in this position at the University of Wittenberg.

- What will happen if we train on this data?
  - Will loss decrease?
  - How will the model learn to “behave”?

Annotation Artifacts, Reasoning Shortcuts: NLI

Reminder: NLI with BERT

entailed/neutral/contradiction



Devlin et al. (2019)

NLI: Hypothesis-only Baselines

Premise: *A woman on a deck is selling bamboo sticks.*

Label?

Hypothesis: *A man is selling bamboo sticks*

Hypothesis: *A man is juggling flaming chainsaws*

Hypothesis: *Eighteen flying monkeys are in outer space*

- Not all of these things have the same likelihood of being true a priori
- What might the model learn to do in this case?

NLI: Hypothesis-only Baselines

Premise	A woman selling bamboo sticks talking to two men on a loading dock.
Entailment	There are **at least** three **people** on a loading dock.
Neutral	A woman is selling bamboo sticks **to help provide for her family.**
Contradiction	A woman is **not** taking money for any of her sticks.

- What's different about this neutral sentence?
  - To create neutral sentences: annotators *add information*
- What's different about this contradictory sentence?
  - To create contradictions: annotators *add negation*
- These are not broadly representative of what can happen in other settings. There is no "natural" distribution of NLI, but this is still very restrictive

NLI: Hypothesis-only Baselines

Premise	A woman selling bamboo sticks talking to two men on a loading dock.
Entailment	There are **at least** three **people** on a loading dock.
Neutral	A woman is selling bamboo sticks **to help provide for her family.**
Contradiction	A woman is **not** taking money for any of her sticks.

- Models can detect new information or negation easily
- Models can do very well ***without looking at the premise***

Performance of models that only look at the hypothesis: ~70% on 3-class SNLI dataset

	Hyp-only model	Majority class	
SNLI	69.17	33.82	+35.35
MNLI-1	55.52	35.45	+20.07
MNLI-2	55.18	35.22	+19.96

Gururangan et al. (2018); Poliak et al. (2018)

NLI: Heuristics (HANS)

Heuristic	Definition	Example
Lexical overlap	Assume that a premise entails all hypotheses constructed from words in the premise	**The doctor** was **paid** by **the actor**. $\xrightarrow[\text{WRONG}]{}$ The doctor paid the actor.
Subsequence	Assume that a premise entails all of its contiguous subsequences.	The doctor near **the actor danced**. $\xrightarrow[\text{WRONG}]{}$ The actor danced.
Constituent	Assume that a premise entails all complete subtrees in its parse tree.	If **the artist slept**, the actor ran. $\xrightarrow[\text{WRONG}]{}$ The artist slept.

- Word overlap supersedes actual reasoning in these cases
- They create a test set (HANS) consisting of cases where heuristics like word overlap are misleading. Very low performance

McCoy et al. (2019)

Evidence of Spurious Correlations: Contrast Sets

- How do we control for annotation artifacts? Things like “premises and hypotheses overlap too much” aren’t easy to see!
- For any particular effect like lexical overlap, we could try to annotate data that “breaks” that effect
- Issue: breaking one correlation may just result in another one surfacing. How do we “break” them all at the same time?
- Solution: construct new examples through *minimal edits that change the label.*

Gardner et al. (2020)

Evidence of Spurious Correlations: Contrast Sets

Hardly one to be faulted for his ambition or his vision, it is genuinely unexpected, then, to see all Park's effort add up to so very little. . . . The premise is promising, gags are copious and offbeat humour abounds but it all fails miserably to create any meaningful connection with the audience.
(Label: Negative)

Hardly one to be faulted for his ambition or his vision, **here we see all Park's effort come to fruition.** . . . The premise is **perfect**, gags are **hilarious** and offbeat humour abounds, **and it creates a deep** connection with the audience.
(Label: Positive)

- By minimally editing an example, we control for pretty much all of the possible shortcuts that apply to the original.
- E.g., [summary starts with "*Hardly*" -> negative] is a pattern that could not hold anymore

Gardner et al. (2020)

Evidence of Spurious Correlations: Contrast Sets

Dataset	# Examples	# Sets	Model	Original Test	Contrast	
NLVR2	994	479	LXMERT	76.4	61.1	(–15.3)
IMDb	488	488	BERT	93.8	84.2	(–9.6)
MATRES	401	239	CogCompTime2.0	73.2	63.3	(–9.9)
UD English	150	150	Biaffine + ELMo	64.7	46.0	(–18.7)
PERSPECTRUM	217	217	RoBERTa	90.3	85.7	(–4.6)
DROP	947	623	MTMSN	79.9	54.2	(–25.7)

Gardner et al. (2020)

Solutions

Broad Solutions

- Most solutions involve changing what data is trained on
  - Subset of data
  - Soft subset (i.e., reweight the existing examples)
  - Superset: add adversarially-constructed data, contrast sets, etc.
- For subsets: what do we train on?
  - Don’t train on stuff that allows you to cheat
  - Train on examples that teach the real task rather than shortcuts

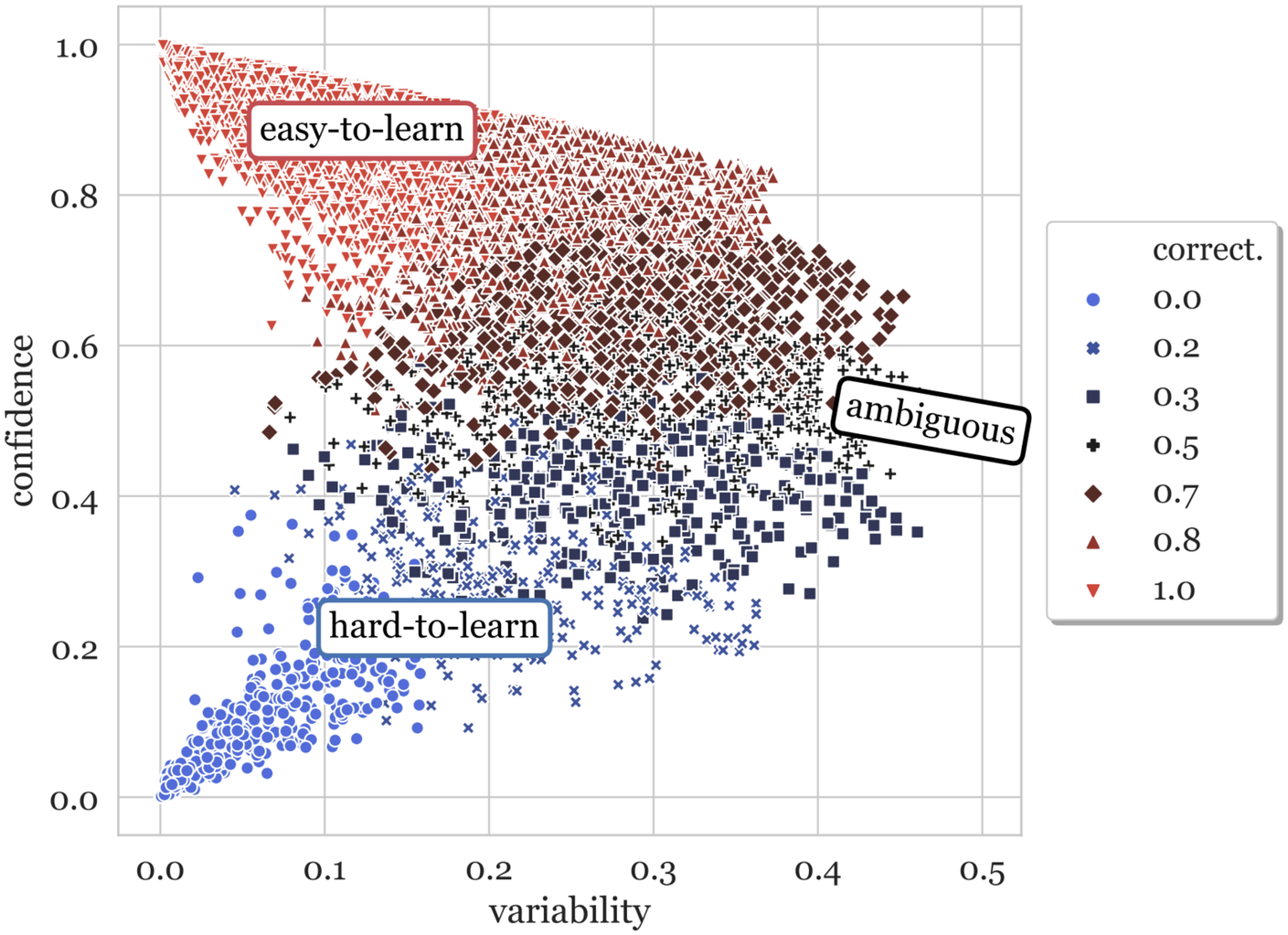
Dataset Cartography

- What happens with each particular example during training?
- Spurious correlations are *easy to learn*: a model should learn these early and always get them right
- Imagine a very challenging example
  - Model prediction may change a lot as it learns this example, may be variable in its predictions
- Imagine a mislabeled example
  - Probably just always wrong unless it gets overfit

Swayamdipta et al. (2021)

Data Maps

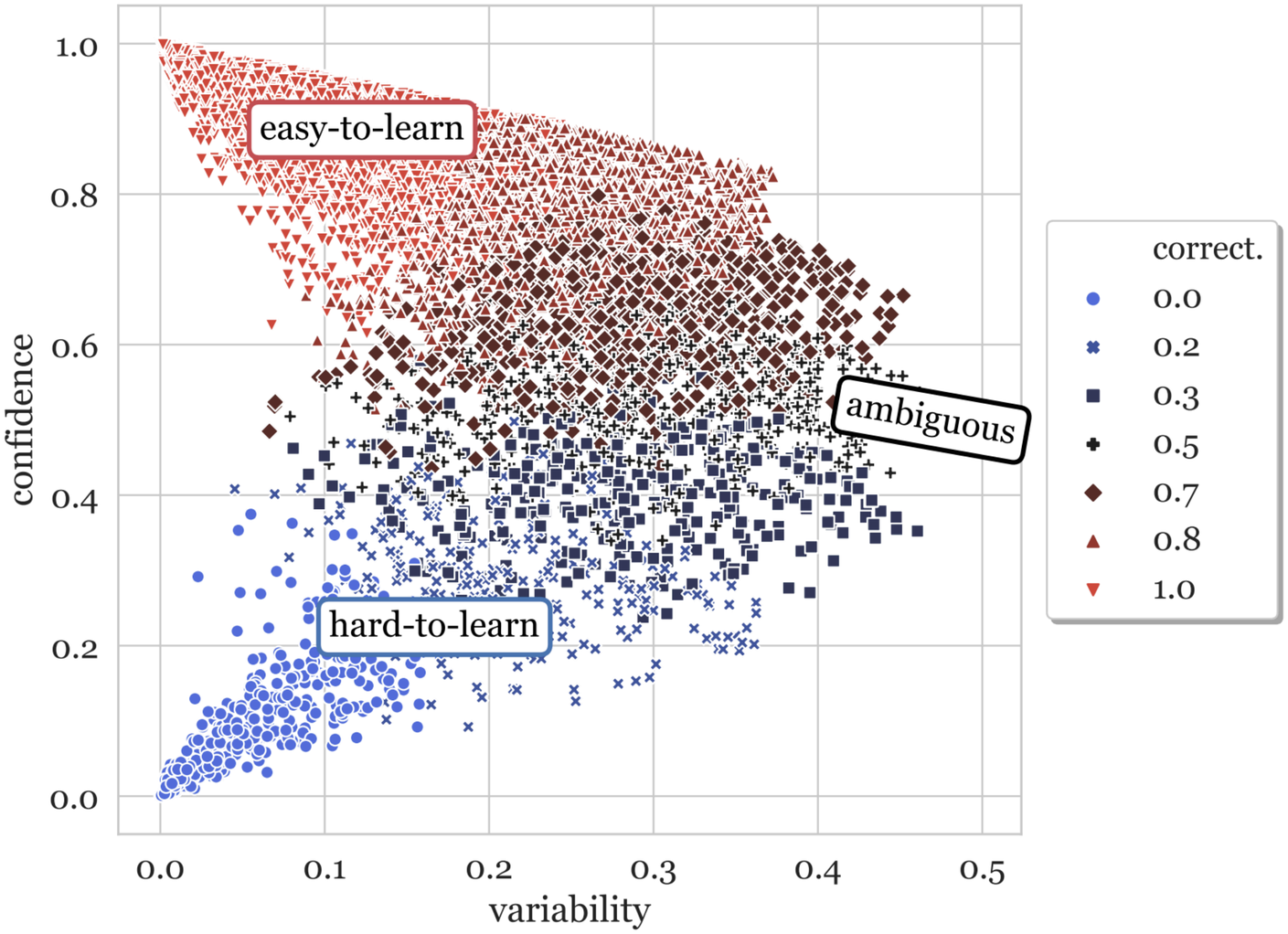
- Confidence: mean probability of correct label
- Variability: standard deviation in probability of the correct label
- Ambiguous examples: possible learnable (model knows it sometimes but not other times), but hard!



Swayamdipta et al. (2021)

Data Maps

- What to do with them?
- Training on hard-to-learn or ambiguous examples leads to better performance out-of-domain

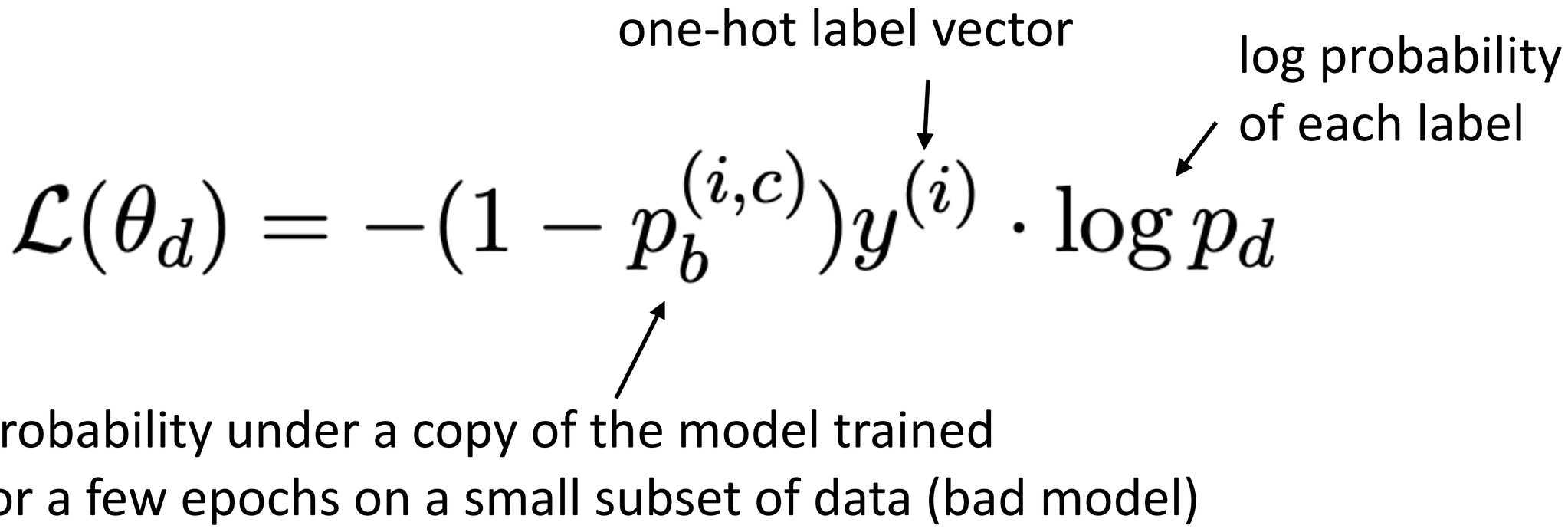


Swayamdipta et al. (2021)

Debiasing

- Other ways to identify easy examples other than data maps

- Train some kind of a weak model and discount examples that it fits easily



Utama et al. (2020)

Debiasing

Method	MNLI (Acc.)		
	dev	*HANS*	Δ
BERT-base	84.5	61.5	-
Reweighting known-bias	83.5‡	69.2‡	+7.7
Reweighting self-debias	81.4	68.6	+7.1
Reweighting ♠ self-debias	82.3	69.7	**+8.2**

- On the challenging HANS test set for NLI, this debiasing improves performance substantially
- In-domain MNLI performance goes down

Utama et al. (2020)

Debiasing

- Other work has explored similar approaches using a known bias model

$$\hat{p}_i = softmax(\log(p_i) + \log(b_i))$$

probabilities from learned bias model — like the weak model from Utama et al. (prev. slides), but you define its structure

- *Ensembles* the weak model with the model you actually learn.
- Your actual model learns the *residuals* of the weak model: the difference between the weak model's output distribution and the target distribution.
- This lets it avoid learning the weak model's biases!

He et al. (2019), Clark et al. (2019)

Takeaways

- Strong neural models trained on “tough” datasets may fail to generalize because they learn annotation artifacts
- By reweighting data or changing the training paradigm, you can learn a model that generalizes better
- Most gains will show up **out-of-domain**. Very hard to get substantial improvements on the same dataset, unless you consider small subsets of examples (e.g., the toughest 1% of examples by some measure)
- As more “generalist” LLMs are learned, this problem goes away…but there’s always a tradeoff when you want to fine-tune them for certain tasks
- Next time: further understanding in-context learning