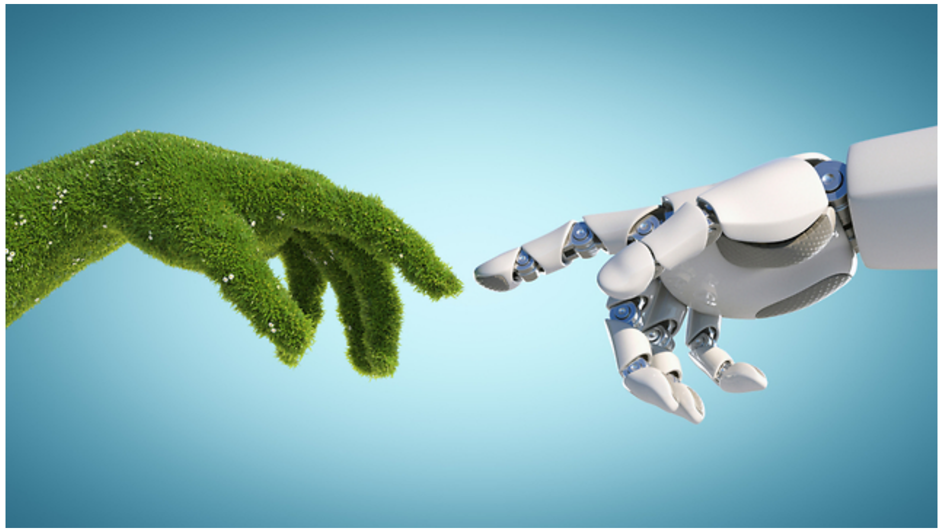
Green NLP

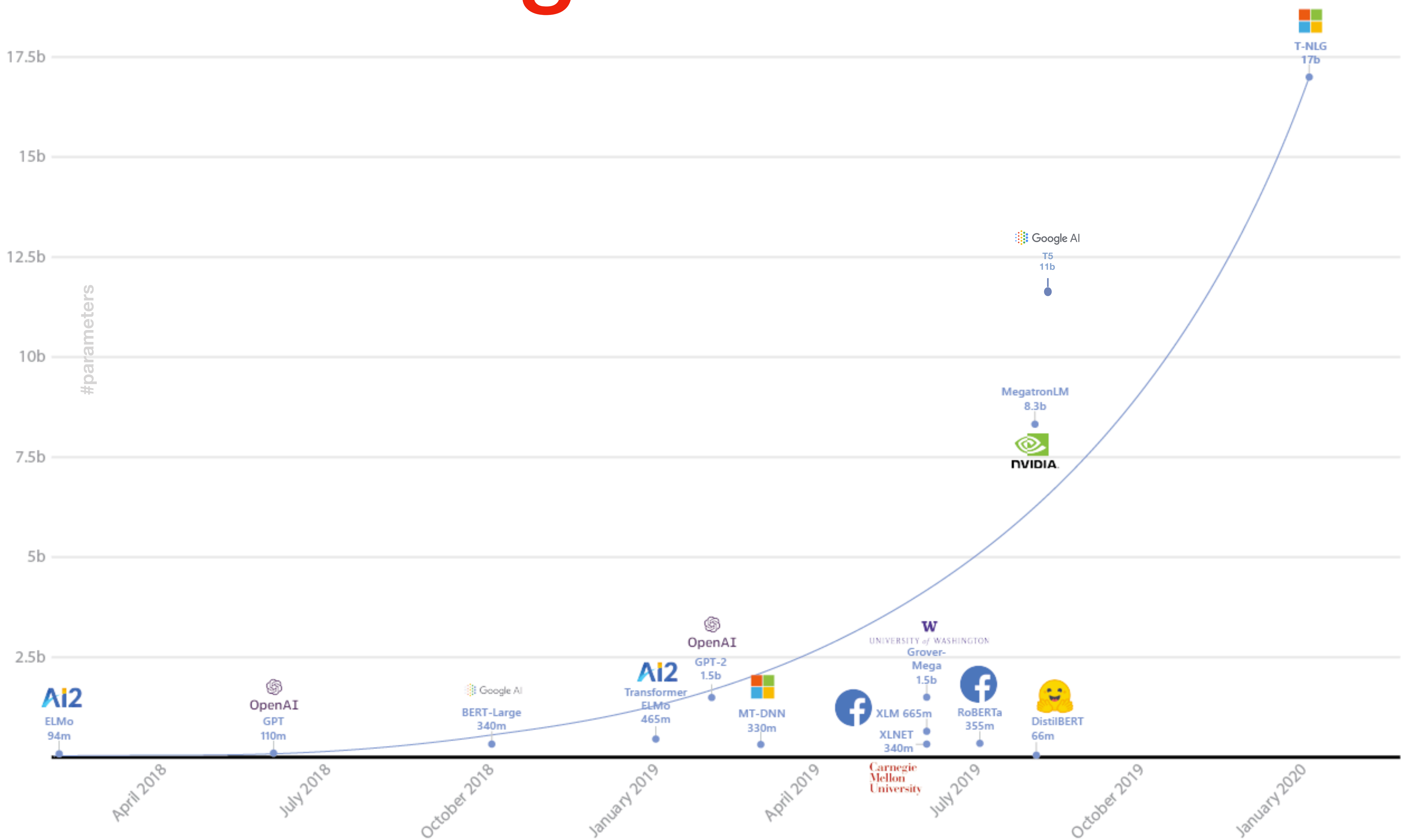
Roy Schwartz

Allen Institute for AI/
University of Washington → Hebrew University
of Jerusalem



THE HEBREW
UNIVERSITY
OF JERUSALEM

Premise:
Big Models



https://www.microsoft.com/en-us/research/blog/turing-nlg-a-17-billion-parameter-language-model-by-microsoft/

https://medium.com/huggingface/distilbert-8cf3380435b5

Problems with Big Models

Research community

Synced

AI TECHNOLOGY & INDUSTRY REVIEW

FEATURE INDUSTRY TECHNOLOGY COMMUNITY ABOUT US REPORT CONTRIBUTE TO SYNCED REVIEW

https://syncedreview.com/2019/06/27/the-staggering-cost-of-training-sota-ai-models/

Problems with Big Models

General AI Community

NCE MACHINE LEARNING PROGRAMMING VISUALIZATION AI PICKS MO

Too big to deploy: How GPT-2 is breaking servers

A look at the bottleneck around deploying massive models to production

Caleb Kaiser Follow

Jan 31 · 7 min read

https://towardsdatascience.com/too-big-to-deploy-how-gpt-2-is-breaking-production-63ab29f0897c

Problems with Big Models

Global Community

Consumption	CO_2e (lbs)
Air travel, 1 person, NY↔SF	1984
Human life, avg, 1 year	11,023
American life, avg, 1 year	36,156
Car, avg incl. fuel, 1 lifetime	126,000

Training one model (GPU)	
NLP pipeline (parsing, SRL)	39
w/ tuning & experiments	78,468
Transformer (big)	192
w/ neural arch. search	626,155

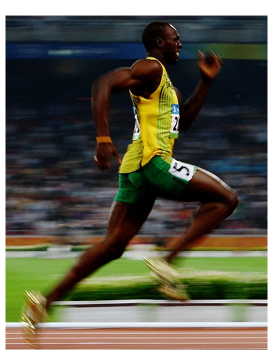
Strubell et al. (2019)

Green AI

Schwartz*, Dodge*, Smith & Etzioni (2019)

- Goals:
  - Enhance reporting of computational budgets

    - Add a *price-tag* for scientific results
  - Promote efficiency as a core evaluation for NLP



    - Inference, training, model selection (e.g., hyperparameter tuning)
    - **In addition to** accuracy

Big Models are Important

- Push the limits of SOTA
- Released large pre-trained models save compute
- Large models are potentially faster to train
    - Li et al. (2020)
- But, big models have **concerning side affects**
    - Inclusiveness, adoption, environment
- Our goal is to **mitigate these side affects**

Outline

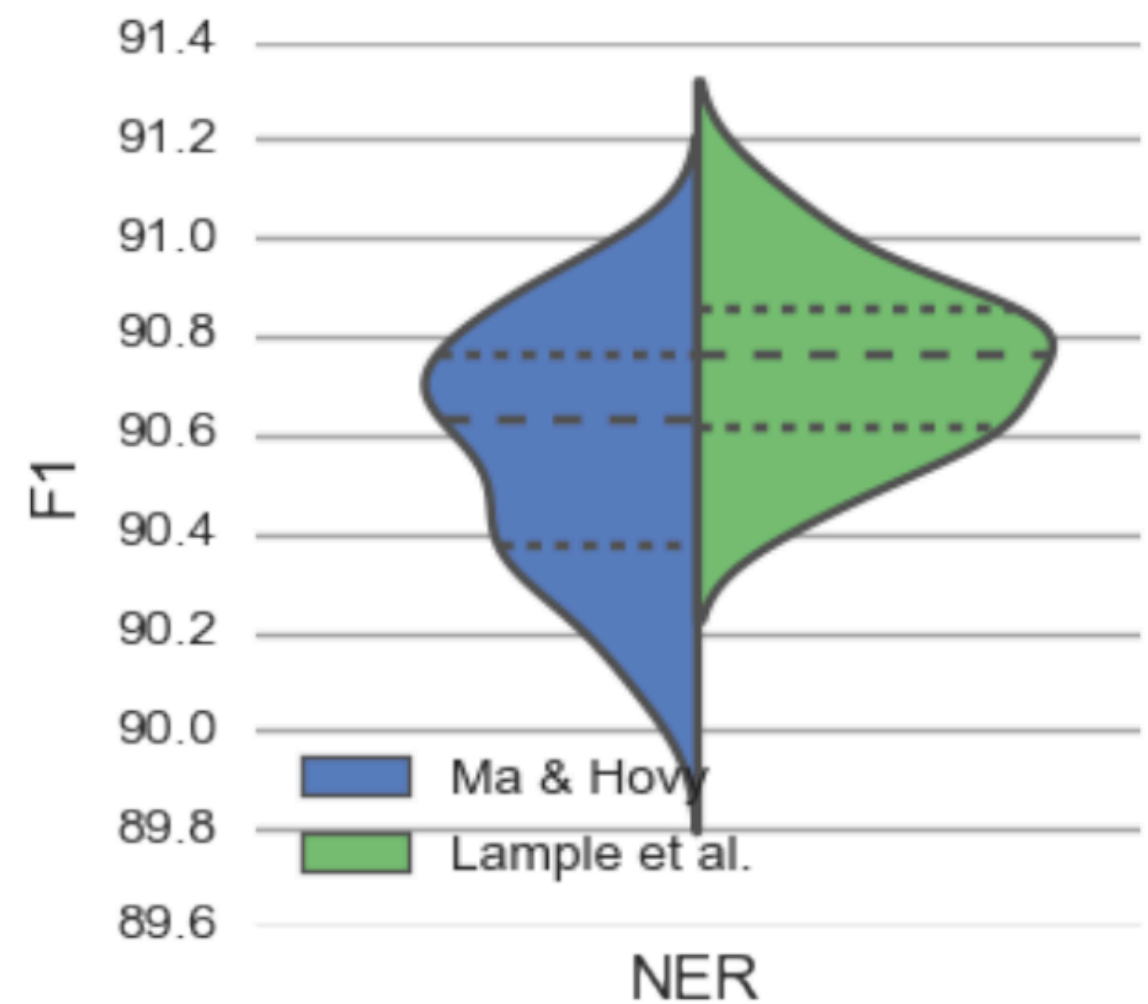
Enhanced Reporting

Efficient Methods

Is Model A > Model B?

Reimers & Gurevych (2017)

Model	F1
Model A	**91.21**
Model B	90.94



Is Model A > Model B?

Melis et al. (2018)

Model	Size	Depth	Valid	Test
Medium LSTM, Zaremba et al. (2014)	10M	2	86.2	82.7
Large LSTM, Zaremba et al. (2014)	24M	2	82.2	78.4
VD LSTM, Press & Wolf (2016)	51M	2	75.8	73.2
VD LSTM, Inan et al. (2016)	9M	2	77.1	73.9
VD LSTM, Inan et al. (2016)	28M	2	72.5	69.0
VD RHN, Zilly et al. (2016)	24M	10	67.9	65.4
NAS, Zoph & Le (2016)	25M	-	-	64.0
NAS, Zoph & Le (2016)	54M	-	-	62.4
AWD-LSTM, Merity et al. (2017) †	24M	3	60.0	57.3
LSTM	10M	1	61.8	59.6
LSTM		2	63.0	60.8
LSTM		4	62.4	60.1
RHN		5	66.0	63.5
NAS		1	65.6	62.7
LSTM	24M	1	61.4	59.5
LSTM		2	62.1	59.6
LSTM		4	60.9	58.3
RHN		5	64.8	62.2
NAS		1	62.1	59.7

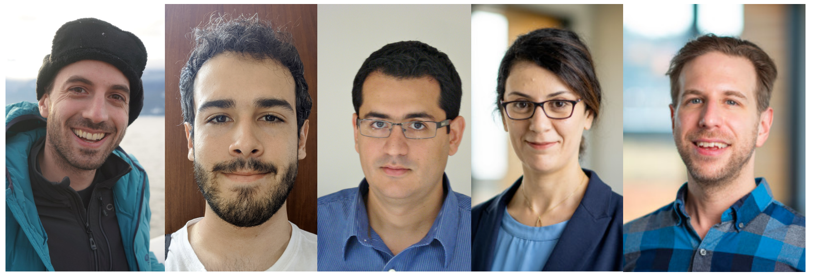
Perplexity (⇓)

Carefully Tuned (1500 trails)

BERT Performs on-par with RoBERTa/ XLNet with better Random Seeds

Dodge, Ilharco, **Schwartz** et al. (2020)

	MRPC	RTE	CoLA	SST
BERT (Phang et al., 2018)	90.7	70.0	62.1	92.5
BERT (Liu et al., 2019)	88.0	70.4	60.6	93.2
BERT (ours)	**91.4**	**77.3**	**67.6**	**95.1**
STILTs (Phang et al., 2018)	90.9	83.4	62.1	93.2
XLNet (Yang et al., 2019)	89.2	83.8	63.6	95.6
RoBERTa (Liu et al., 2019)	90.9	86.6	68.0	96.4
ALBERT (Lan et al., 2019)	90.9	89.2	71.4	96.9



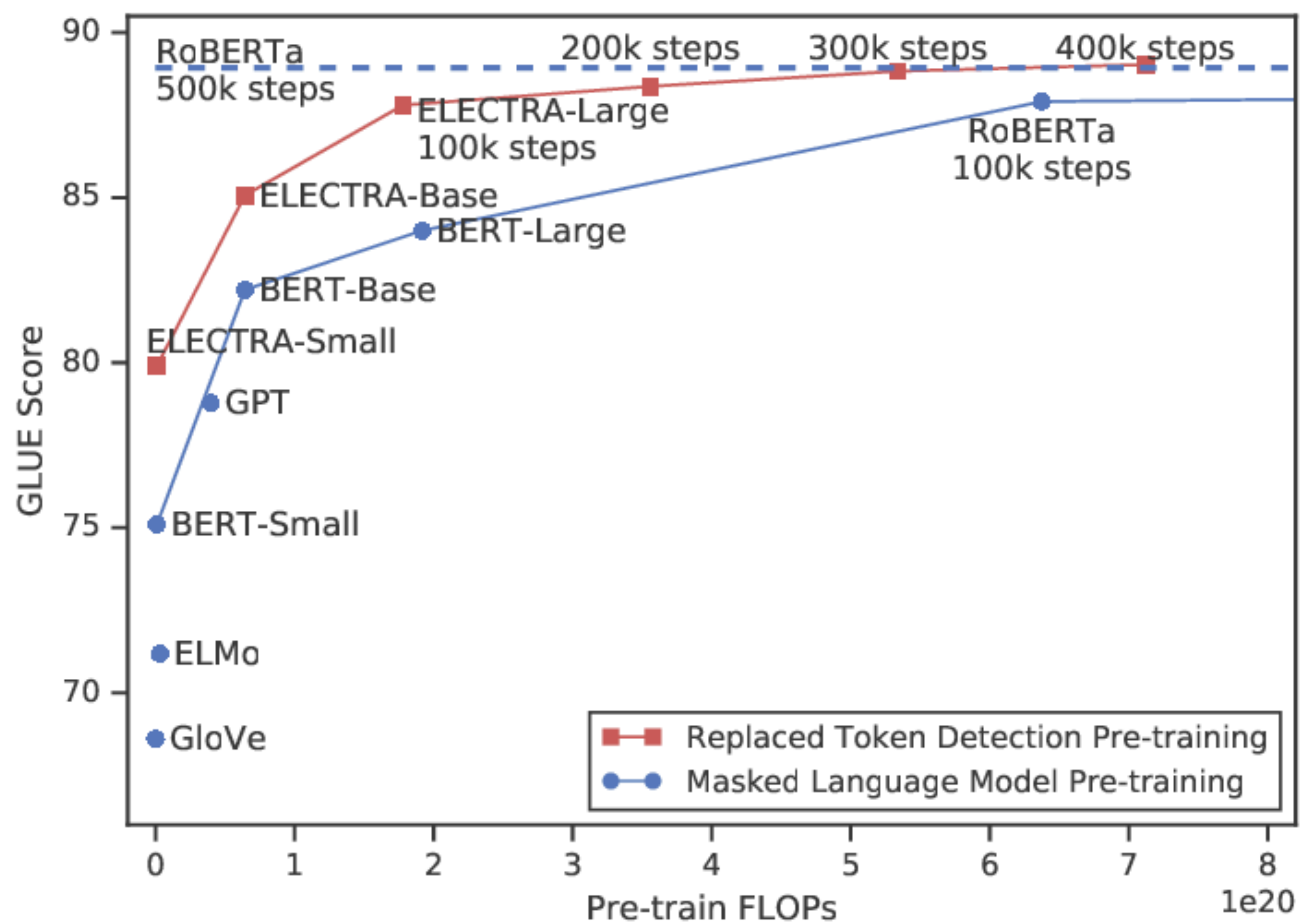
Unfair Comparison

Is **Model A > Model B**?

Better(?) Comparison

Is **Model A > Model B?** | ***Budget***

Budget-Aware Comparison

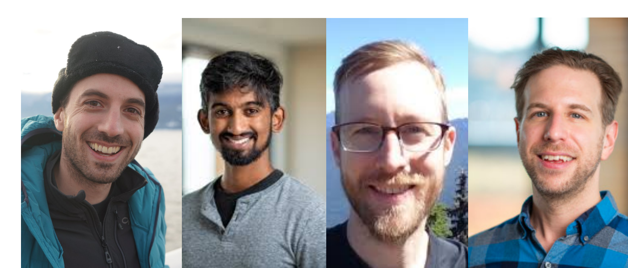


Performance | Budget
(Clark et al., 2020)

Expected Validation

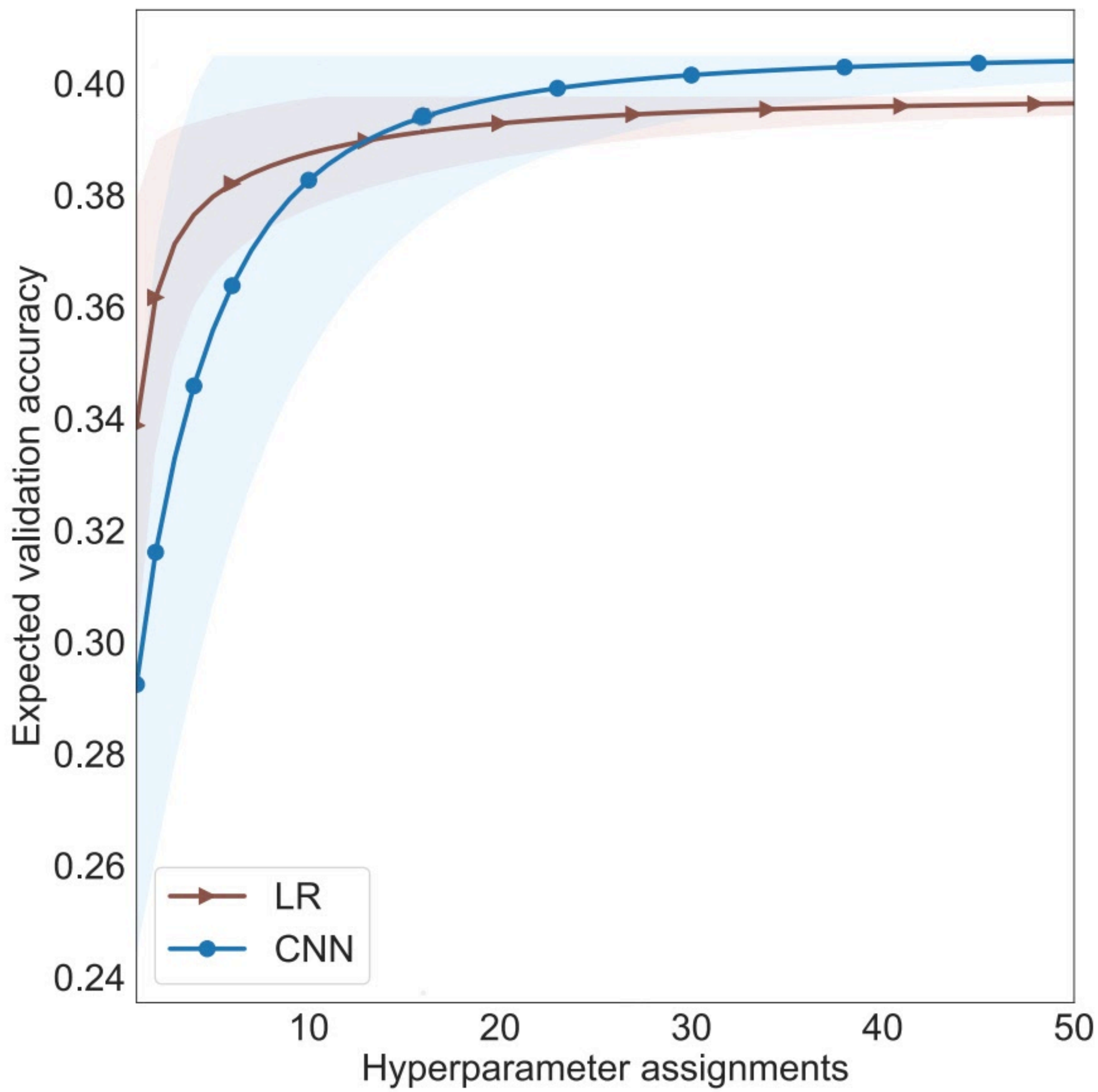
Dodge, Gururangan, Card, **Schwartz** & Smith, 2019

- Input: a set of experimental results $\{V_1, \ldots, V_n\}$
- Define $V_k^* = max_{i \in \{1,\ldots,k\}} V_i$
- **Expected validation performance**: $\mathbb{E}[V_k^* \mid k]$
  - k=1: $mean(\{V_1, \ldots, V_n\})$
  - k=2: $mean(\{max(V_i, V_j) \forall 1 \leq i < j \leq n\})$
  - k=n: $V_n^* = max_{i \in \{1,\ldots,n\}} V_i$



Expected Validation

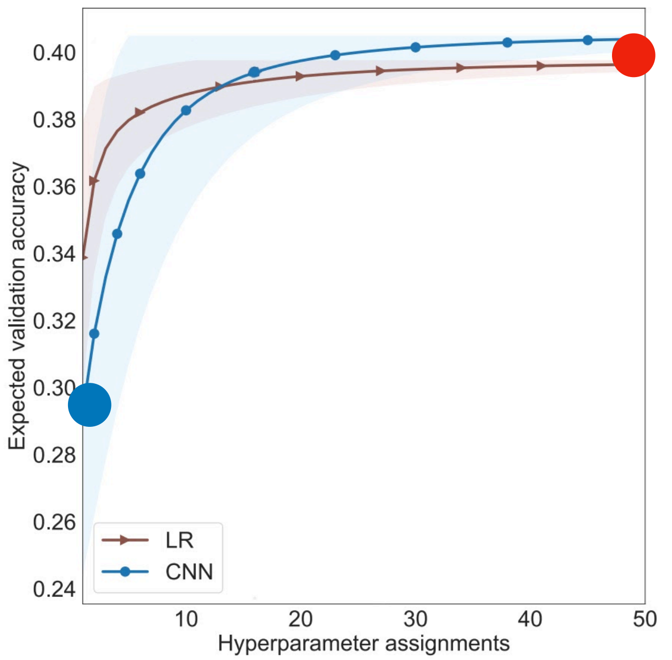
Example: SST5



Expected Validation

Properties

- Doesn’t require rerunning any experiment
    - An analysis of existing results
- More comprehensive than
    - Reporting **max** (the **rightmost** point in our plots)
    - Reporting **mean** (the **leftmost** point in our plots)
- https://github.com/dodgejesse/show_your_work



Reporting
Recap

- Budget-aware comparison
- Expected validation performance
  - Estimation of the amount of computation required to obtain a given accuracy

Reporting

Open Questions

- How much will we gain by pouring **more compute**?
- What should we report?
  - Number of experiments
  - Time
  - FLOPs
  - Energy (KW)
  - Carbon?
- Bigger models, faster training?
  - Li et al. (2020)

Green NLP Goals

Enhanced Reporting

Efficient Methods

Efficient Methods

What are we making more efficient?

Inference

Training

Model Selection

What are we measuring?

Space

Time

Energy

http://mitchgordon.me/machine/learning/2019/11/18/all-the-ways-to-compress-BERT.html
https://blog.inten.to/speeding-up-bert-5528e18bb4ea
https://blog.rasa.com/compressing-bert-for-faster-prediction-2/

Efficient #inference

- **Model distillation** #space; #time; #energy
  - Hinton et al. (2015); MobileBERT (Sun et al., 2019); DistilBERT (Sanh et al., 2019)
- **Pruning** #space / **Structural Pruning** #space; #time; #energy
  - Han et al. (2016); SNIP (Lee et al., 2019); LTH (Frankle & Corbin, 2019)
  - MorphNet (Gordon et al., 2018); Michel et al. (2019); LayerDrop (Fan et al., 2020)
  - Dodge, **Schwartz** et al. (2019)
- **Quantization** #space; #time; #energy
  - Gong et al. (2014); Q8BERT (Zafrir et al., 2019); Q-BERT (Shen et al., 2019)

#space Efficiency

- Weight Factorization
  - ALBERT (Lan et al., 2019); Wang et al., 2019
- Weight Sharing
  - Inan et al., 2016; Press & Wolf, 2017

Early Stopping

#modelselection; #time; #energy

- Stop least promising experiments early on
  - Successive halving (Jamieson & Talwalkar, 2016)
  - Hyperband (Lee et al., 2017)
- Works for random seeds too!
  - Dodge, Ilharco, **Schwartz**, et al. (2020)

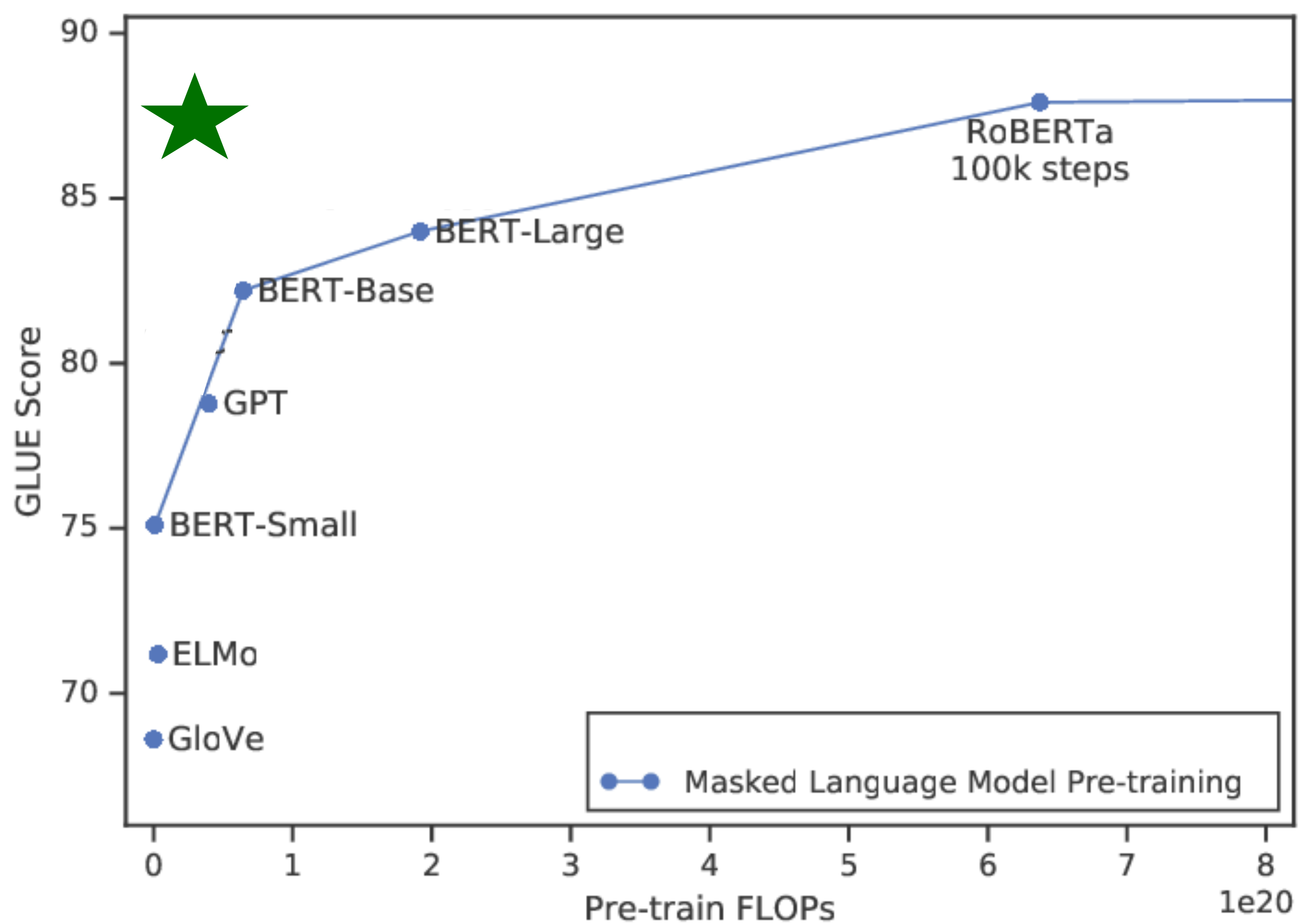
Other Efficient Methods

- Replacing dot-product attention with locally-sensitive hashing
  - #inference; #space; #time; #energy
  - Reformer (Kitaev et al., 2020)
- More efficient usage of the input
  - #inference; #training; #space; #time; #energy
  - ELECTRA (Clark et al., 2020)
- Analytical solution of the backward pass
  - #inference; #space
  - Deep equilibrium model (Bai et al., 2019)

Efficiency/Accuracy Tradeoff

#inference; #time; #energy

Schwartz et al., in review



Performance | Budget
(Clark et al., 2020)

Easy/Hard Instances

Variance in Language

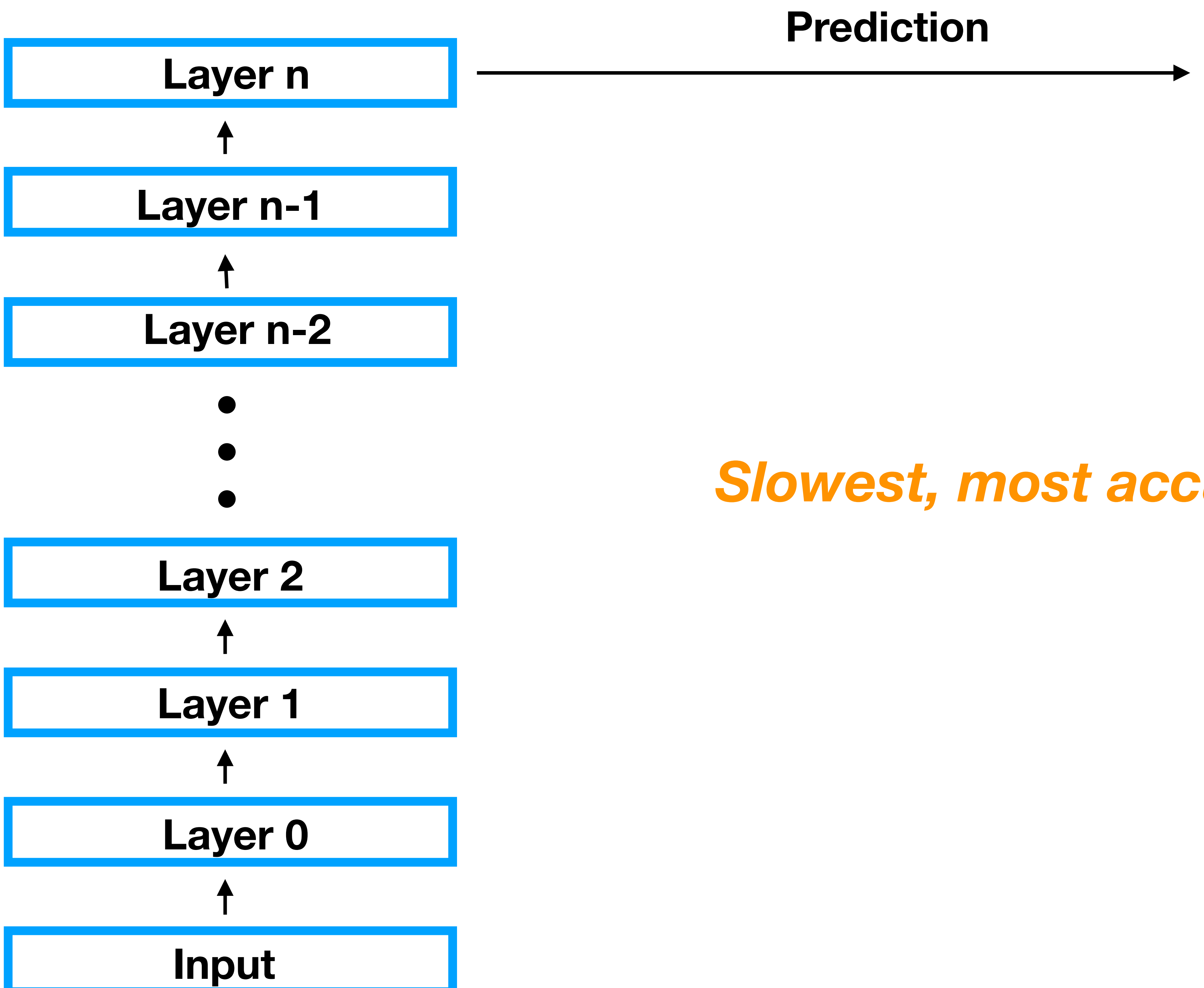
1. *The movie was awesome.*
2. *I could definitely see why this movie received such great critiques, but at the same time I can't help but wonder whether the plot was written by a 12 year-old or by an award-winning writer.*

Matching Model and Instance Complexity

Run an efficient model on "easy" instances,

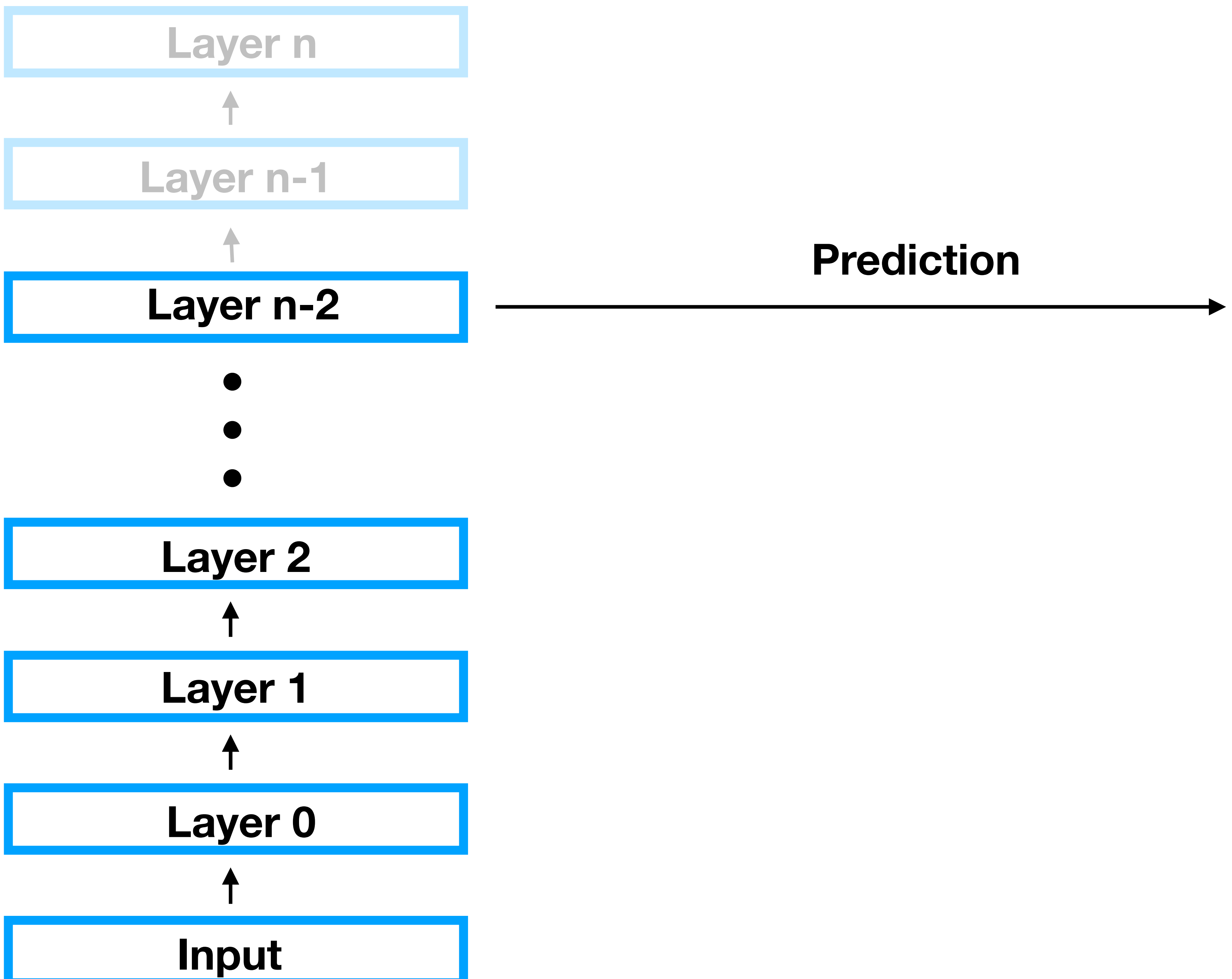
and an expensive model on "hard" instances

Pretrained BERT Fine-tuning

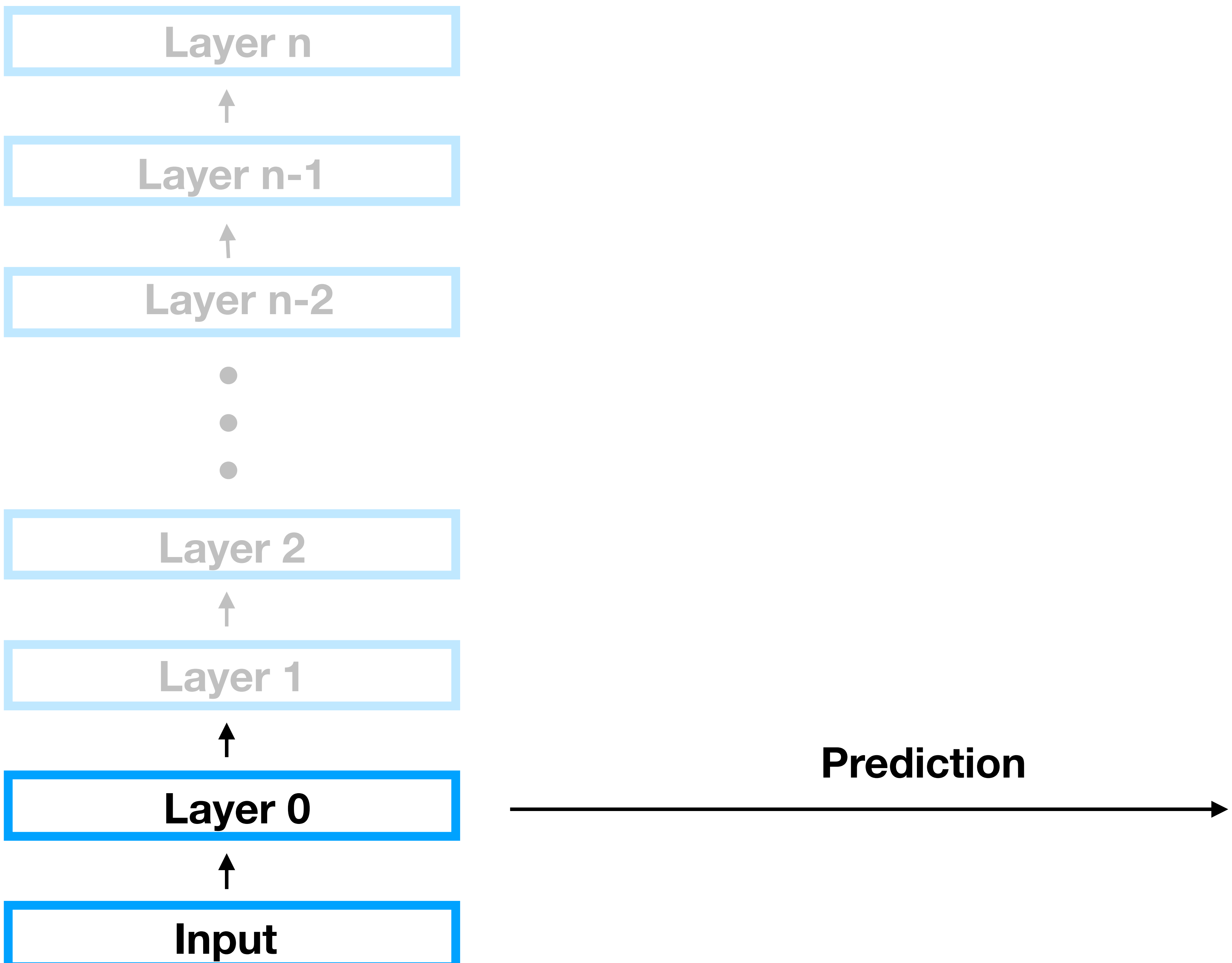


Slowest, most accurate

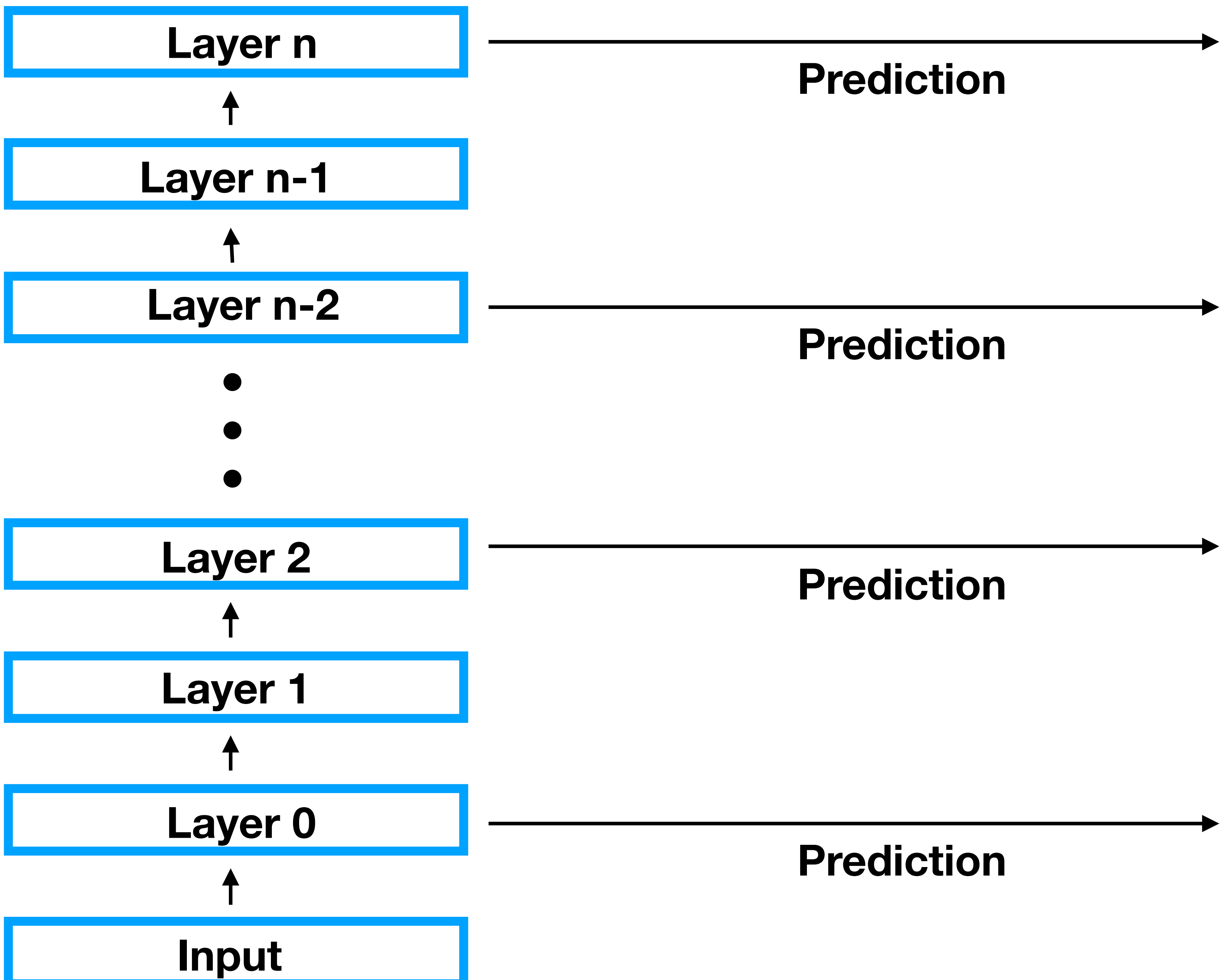
Faster, less Accurate



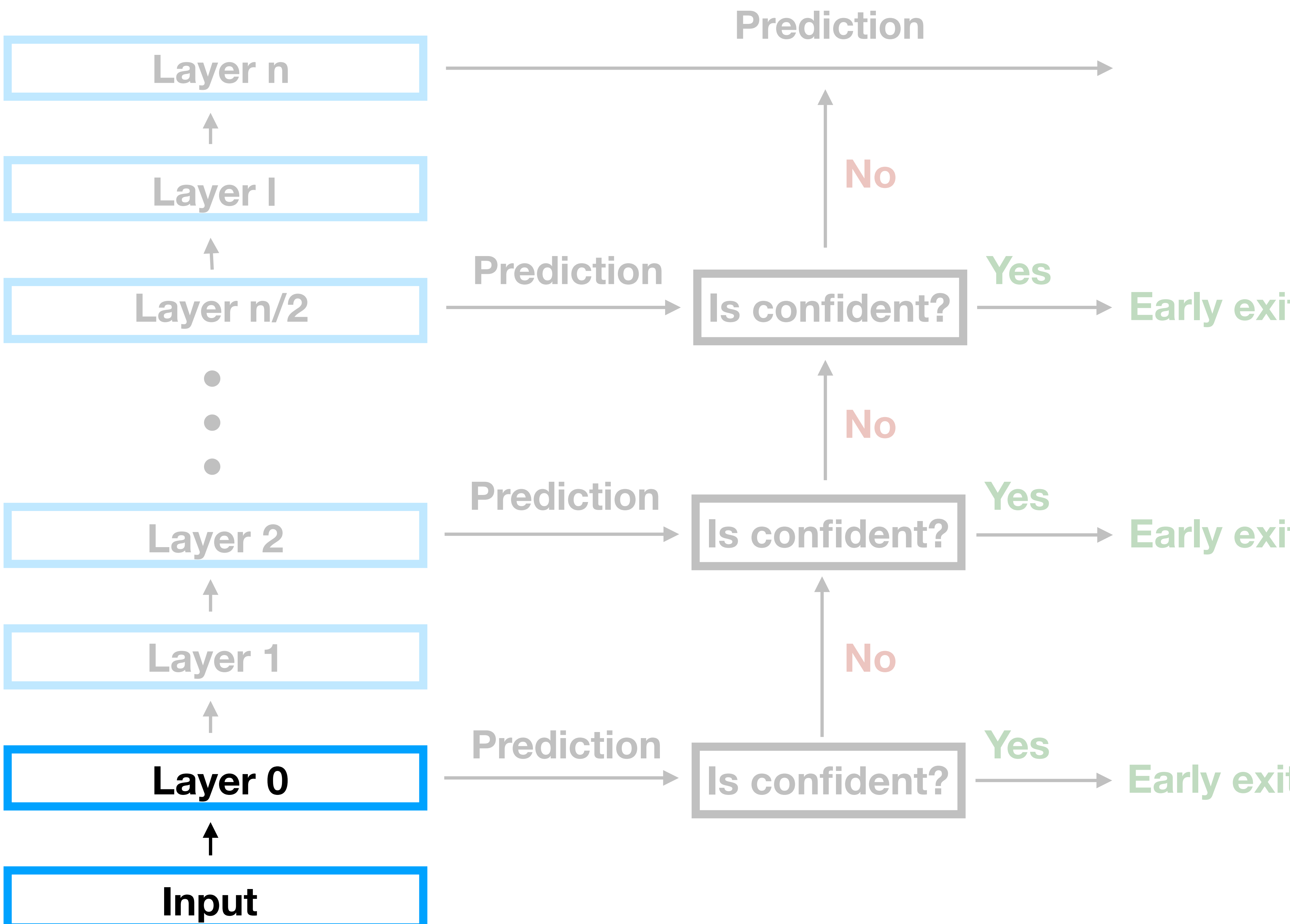
Fastest, least Accurate



Our Approach: **Training Time**



Our Approach: Test Time



Calibrated Confidence Scores

- We interpret the softmax label scores as model confidence

- We **calibrate** our model to encourage the confidence level to correspond to the probability that the model is correct (DeGroot and Fienberg, 1983)

    - We use temperature calibration (Guo et al., 2017)

$$\text{pred} = argmax_i \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)}$$

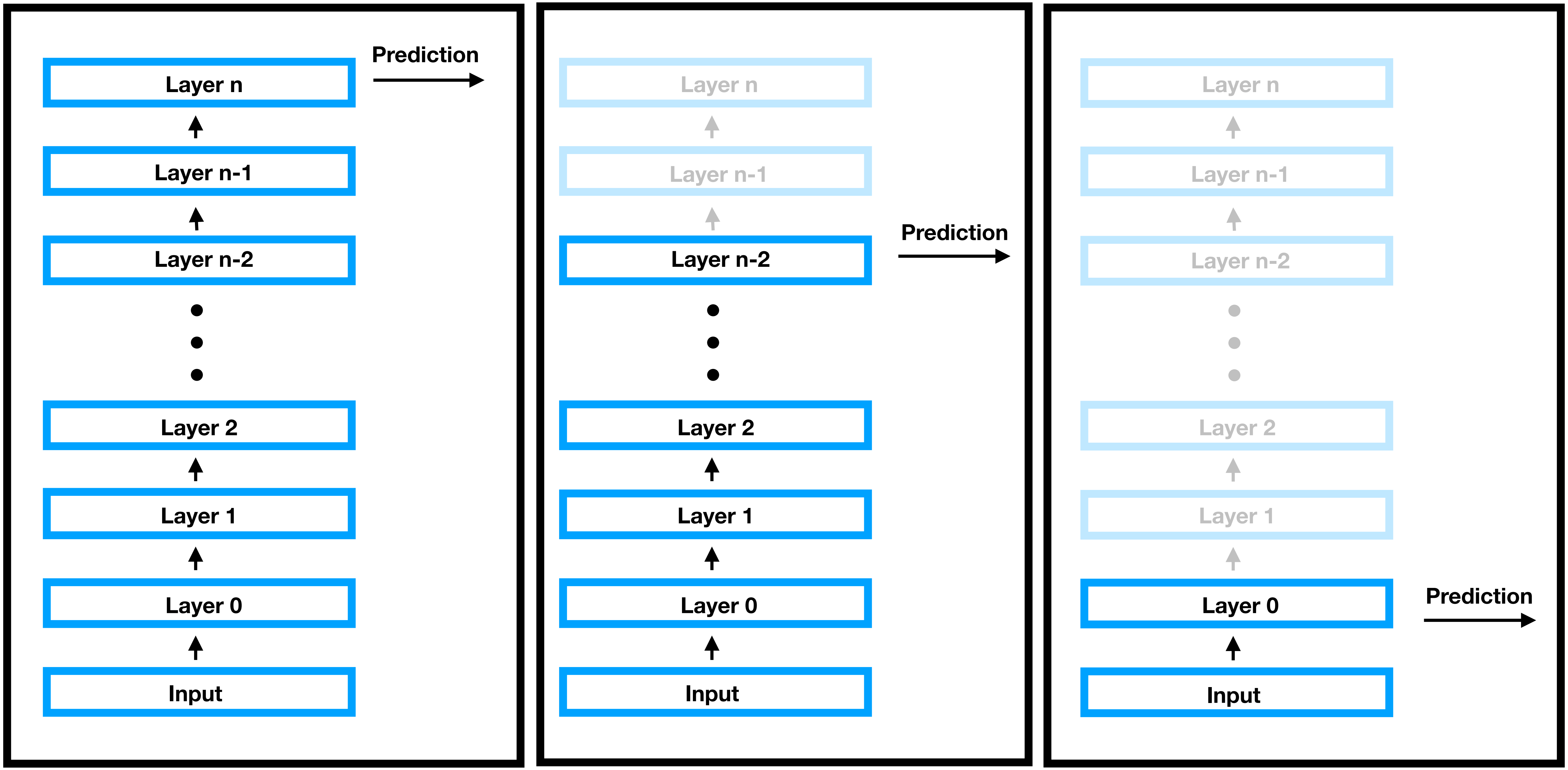
- Speed/accuracy tradeoff controlled by a **single early-exit confidence threshold**

Experiments

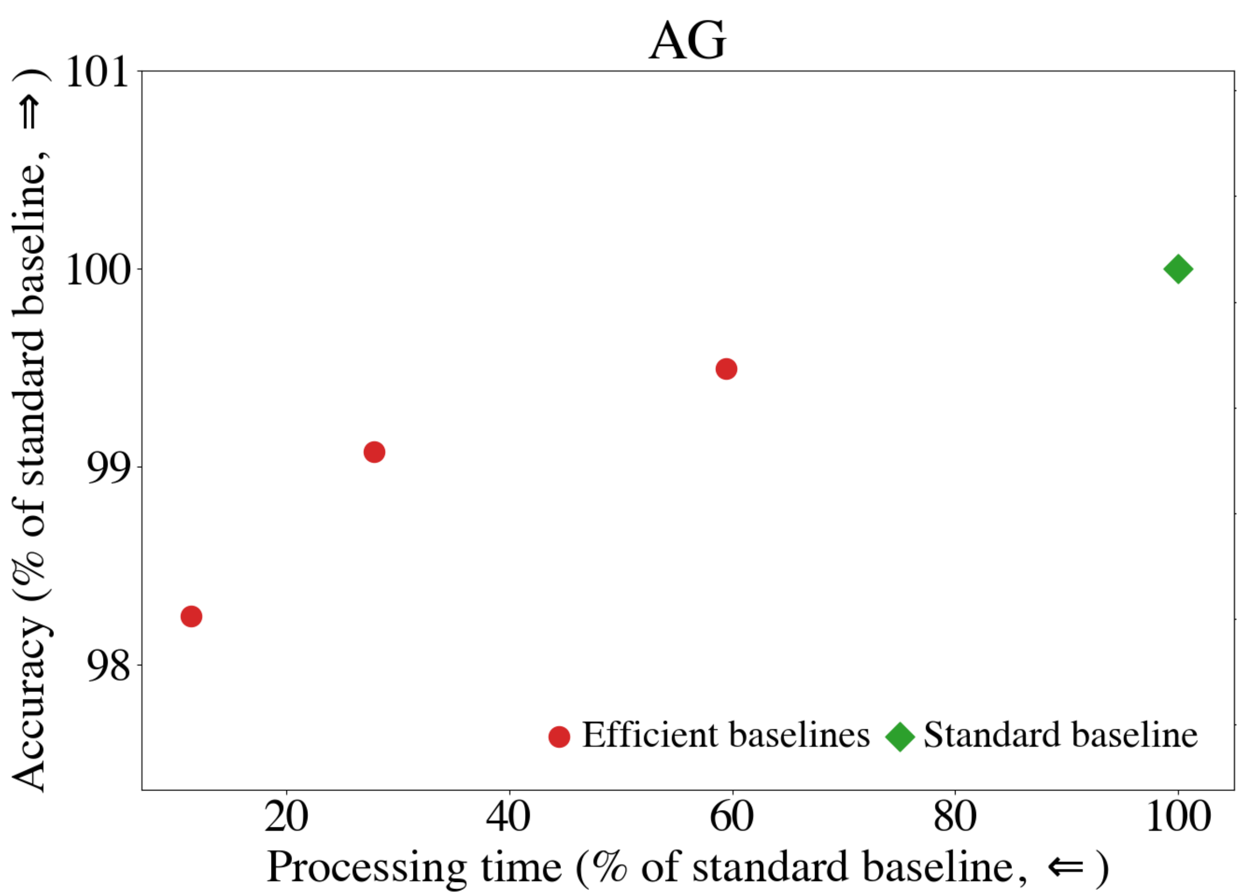
- BERT-large-uncased (Devlin et al., 2019)
  - Output classifiers added to layers 0,4,12 and 23
- Datasets
  - 3 Text classification, 2 NLI datasets

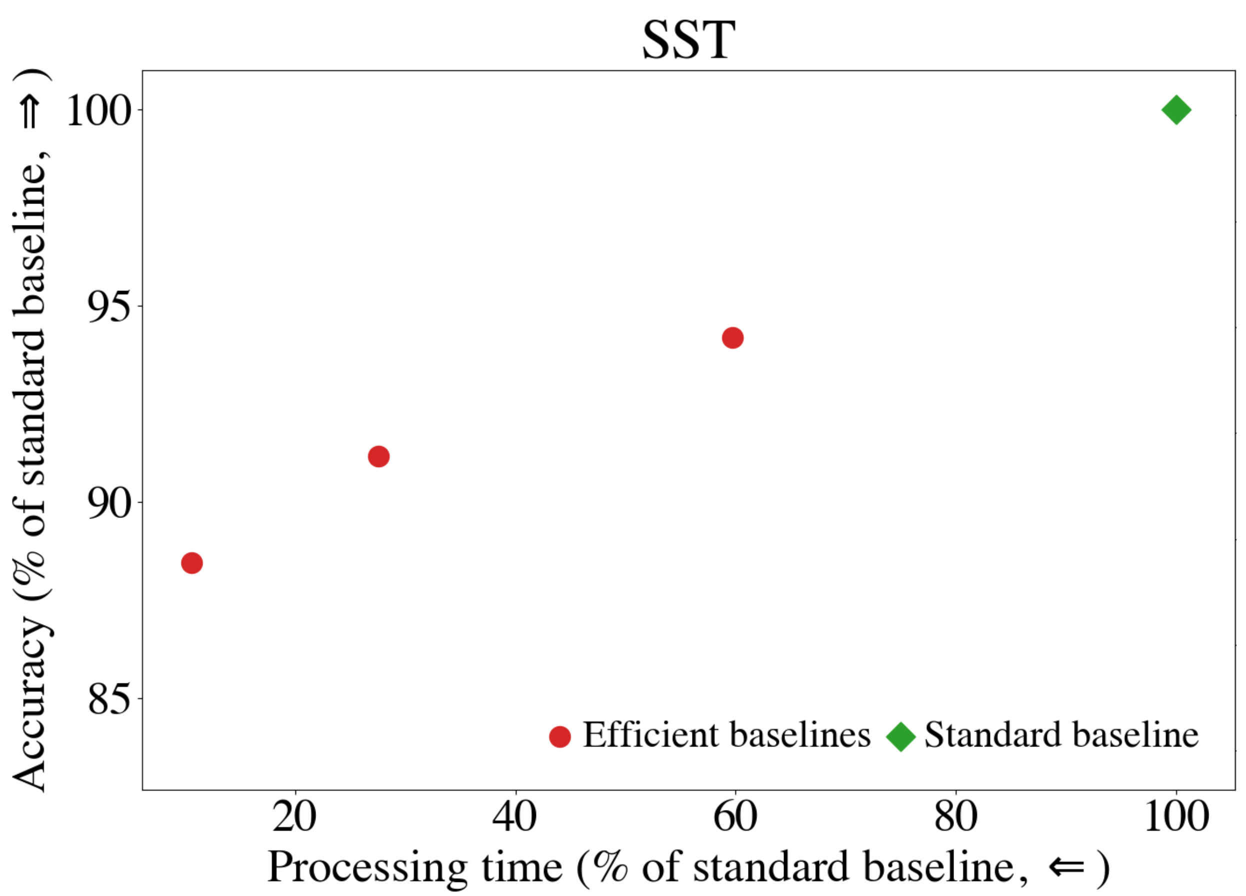
Baselines

Standard baseline **Efficient baselines**

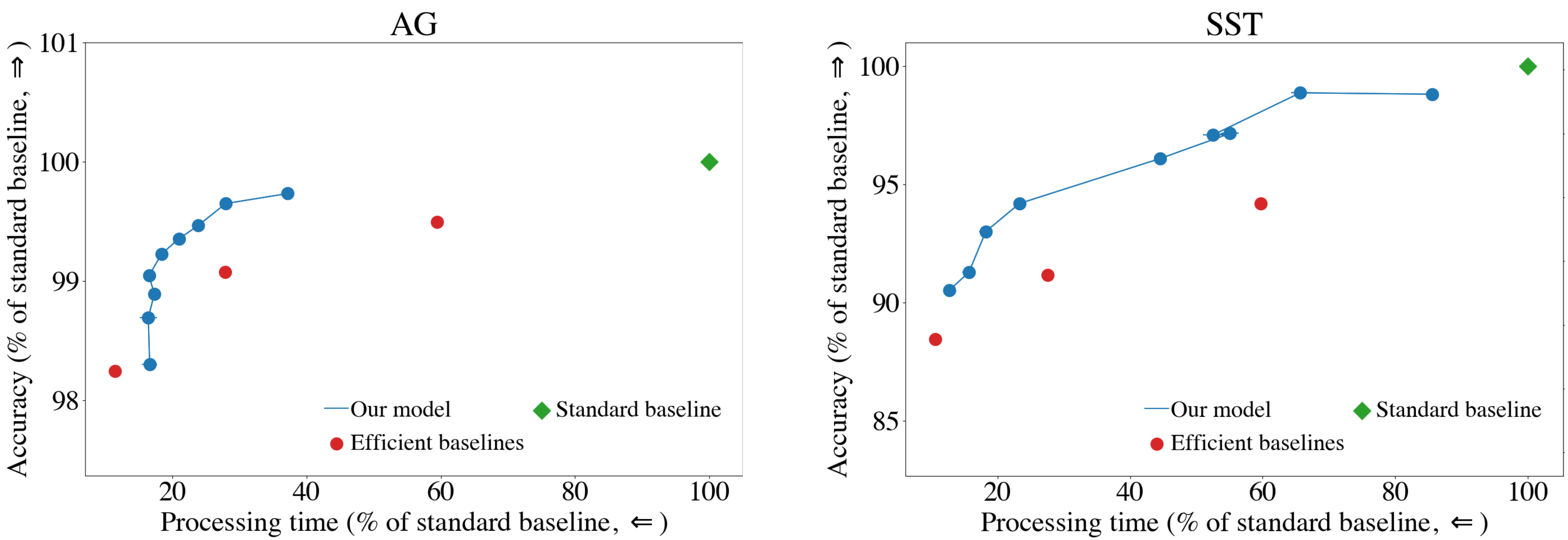


Strong Baselines!

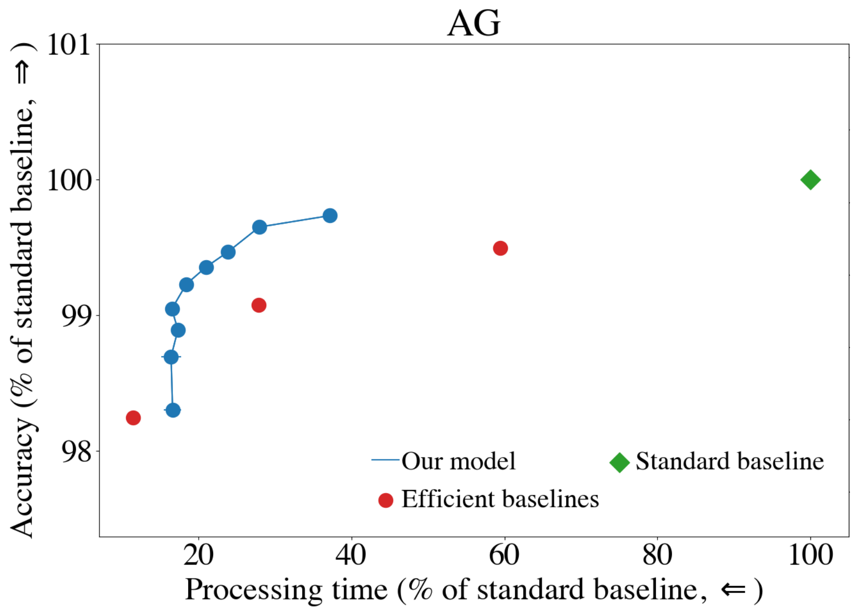


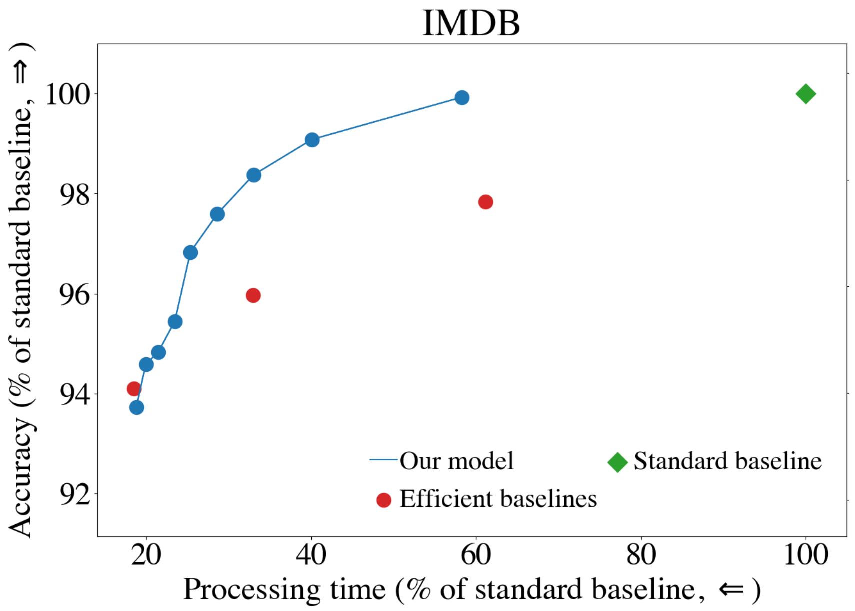


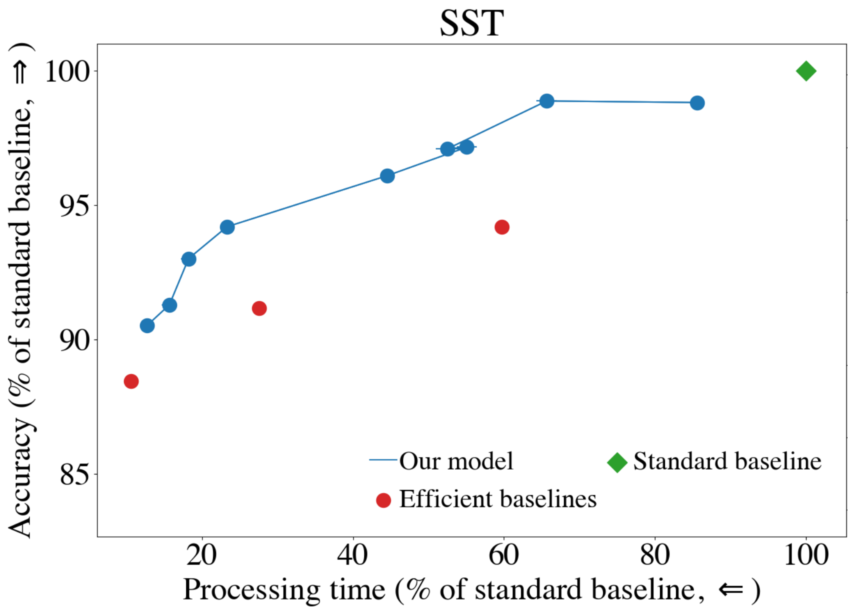
Better Speed/Accuracy Tradeoff

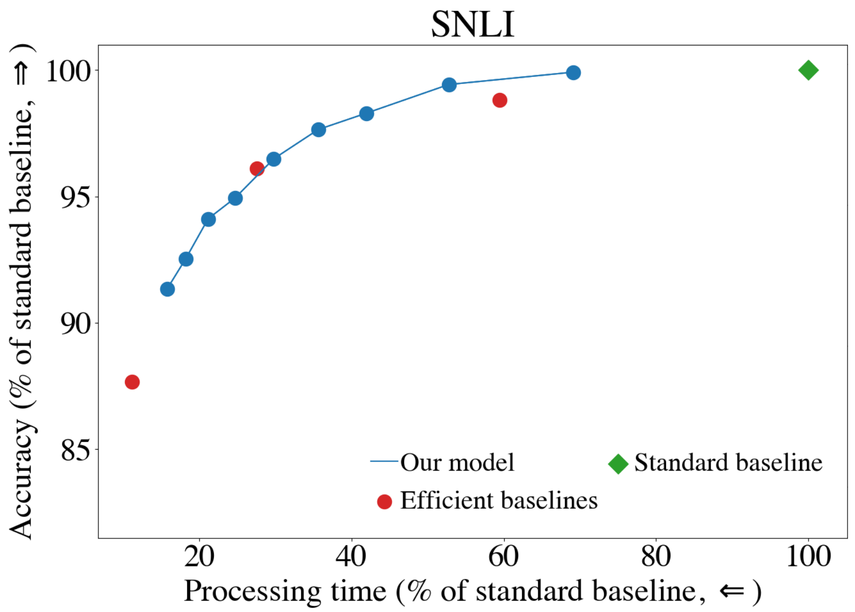


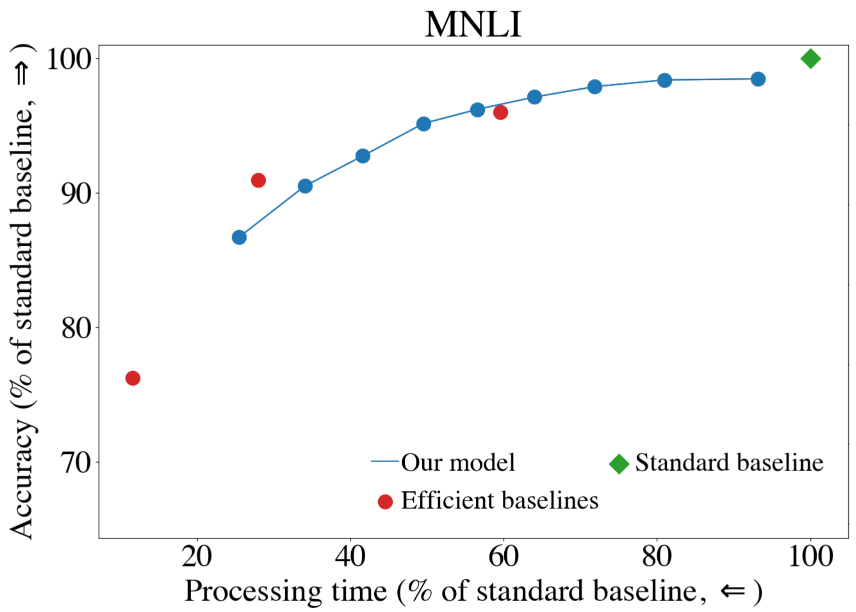
Better Speed-Accuracy Tradeoff











More about our Approach

- No effective growth in parameters
  - < 0.005% additional parameters
- Training is **not** slower
- A **single** trained model provides multiple options along the speed/accuracy tradeoff
  - A single parameter: confidence threshold
- Caveat: requires batch size=1 during inference

Recap

- Efficient inference
- Simple instances exit early, hard instances get more compute
- Training is not slower than the original BERT model
- One model fits all!
  - A single parameter controls for the speed/accuracy curve

Efficiency

Open Questions

- Can we drastically **reduce the price of training BERT**?
- Sample efficiency
- What makes a good sparse structure?
- What makes a good hyperparameter/random seed?

Think Green

- Show your work!
- **Efficiency**, not just accuracy

More about me

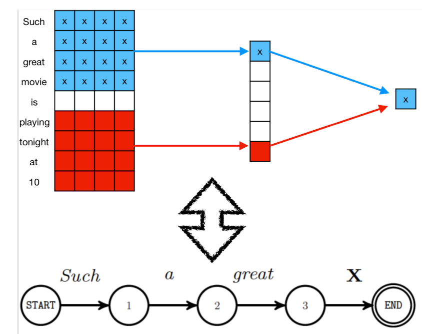
Understanding the NLP Development Cycle

Datasets

Premise	Two dogs are running through a field.
Entailment	There are **animals outdoors**.
Neutral	Some puppies are running **to catch a stick**.
Contradiction	The pets are **sitting on a couch**.

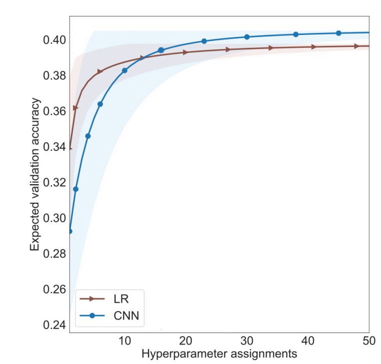
* **Annotation Artifacts (Schwartz et al., 2017; Gururangan et al., 2018)**
* **Inoculation by Fine-Tuning: A Method for Analyzing Challenge Datasets (Liu et al., 2019)**

Models



* **Rational Recurrences (Schwartz et al., 2018; Peng et al., 2018; Merrill et al., in review)**
* **LSTMs Exploit Linguistic Attributes of Data (Liu et al., 2018)**

Experiments



* **Show your Work (Dodge et al., 2019;2020)**

Amazing Collaborators!


Come to
Jerusalem!
Rosemary

Think Green

- **Efficiency** research opportunities

  - Can we drastically **reduce the price of training BERT**?
  - Sample efficiency
  - What makes a good *sparse structure/hyperparameter/random seed*?

- **Reporting** research opportunities

  - How much will we gain by pouring **more compute**?
  - Better reporting methods