Data Efficiency

Advanced NLP: Summer 2023

Anoop Sarkar

Shortformer: Better Language Modeling Using Shorter Inputs

Ofir Press[1,2]   **Noah A. Smith**[1,3]   **Mike Lewis**[2]

[1]Paul G. Allen School of Computer Science & Engineering, University of Washington
[2]Facebook AI Research
[3]Allen Institute for AI
`ofirp@cs.washington.edu`

https://aclanthology.org/2021.acl-long.427/

ShortFormer

- **Staged Training** aka Curriculum Learning

  - Training on shorter subsequences (before moving on to longer ones) leads to faster and more memory-efficient training. Also improves perplexity.

- **Position infused Attention**

  - Transformer XL used cached previously evaluated sequences using *relative* position embeddings to scale to longer inputs

  - Instead of adding absolute position embeddings to the tokens, add positional embeddings directly into the attention layer (keys and queries)

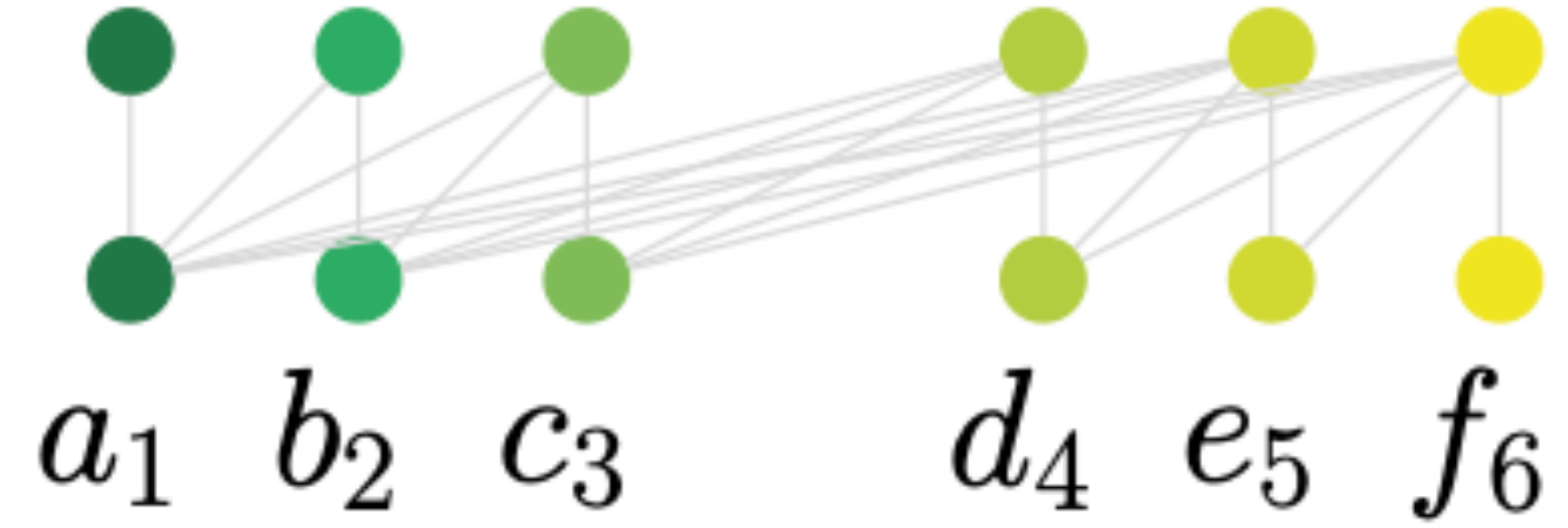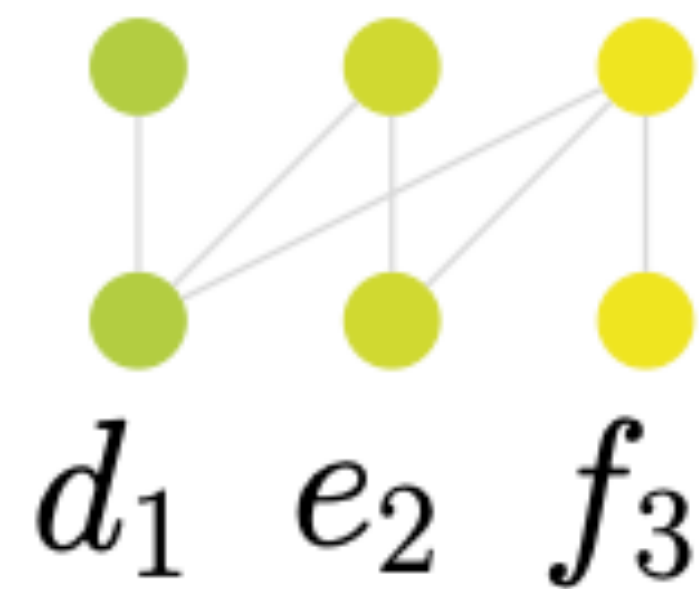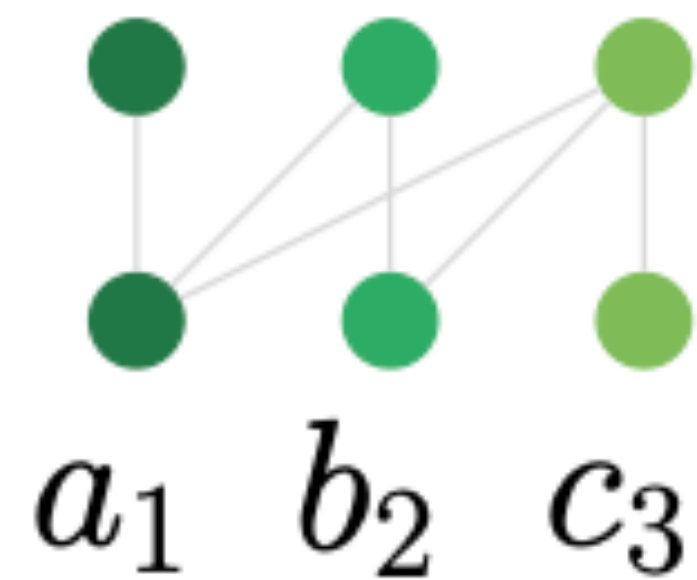 - Get rid of position embeddings at the token level
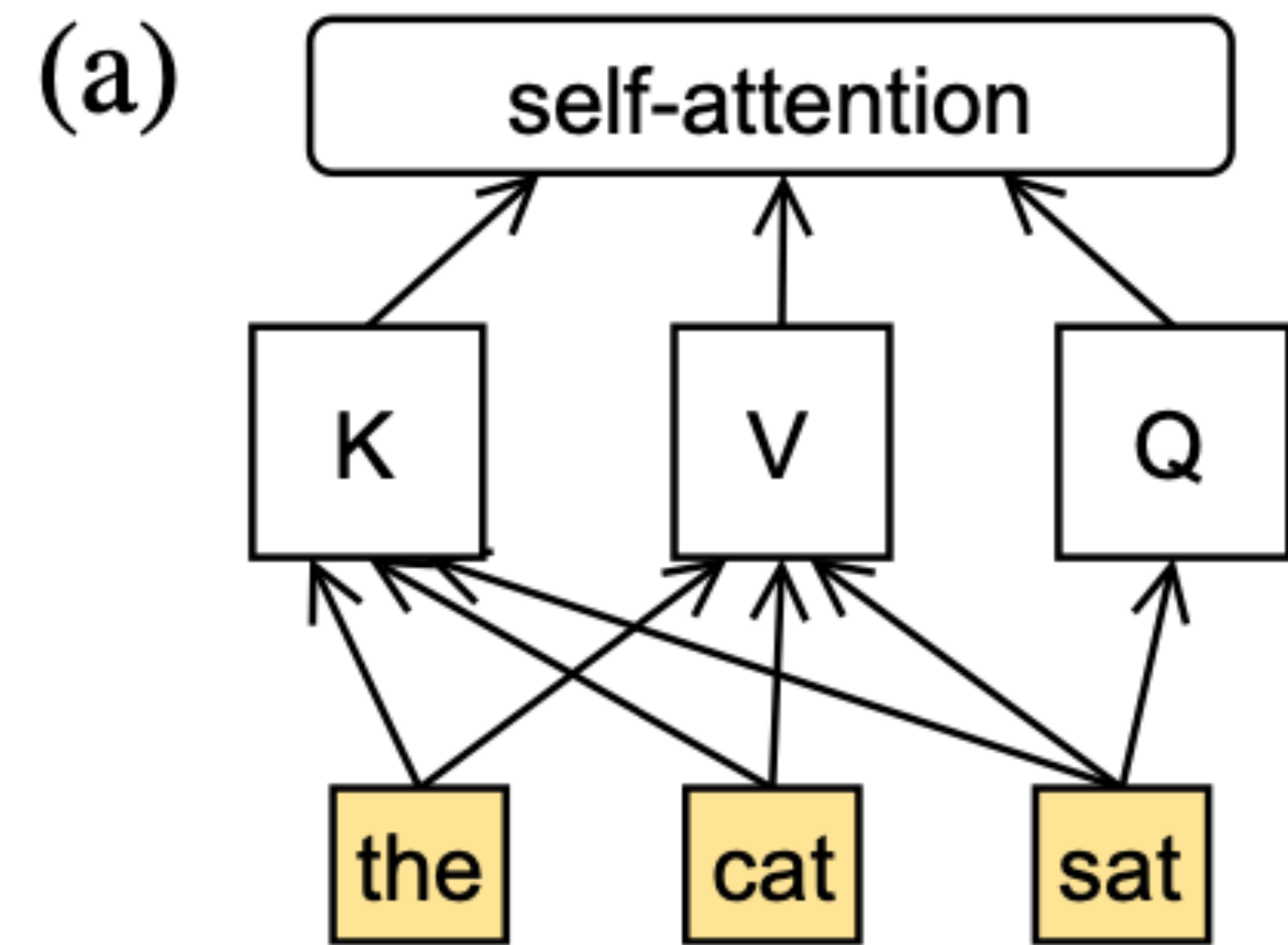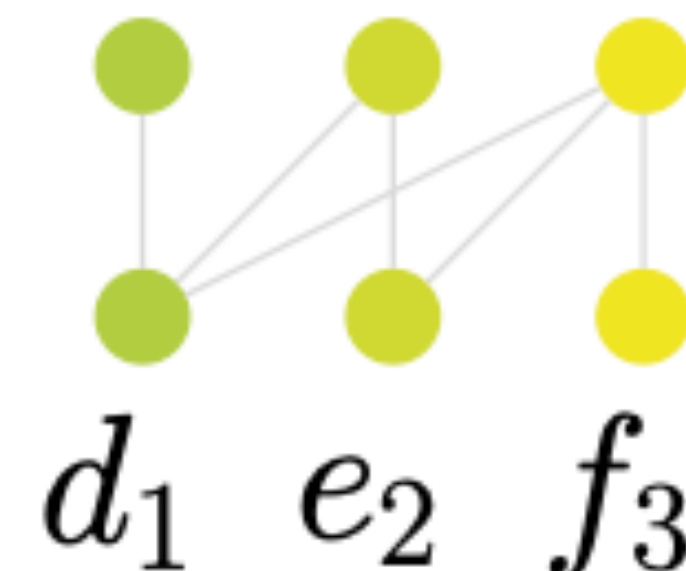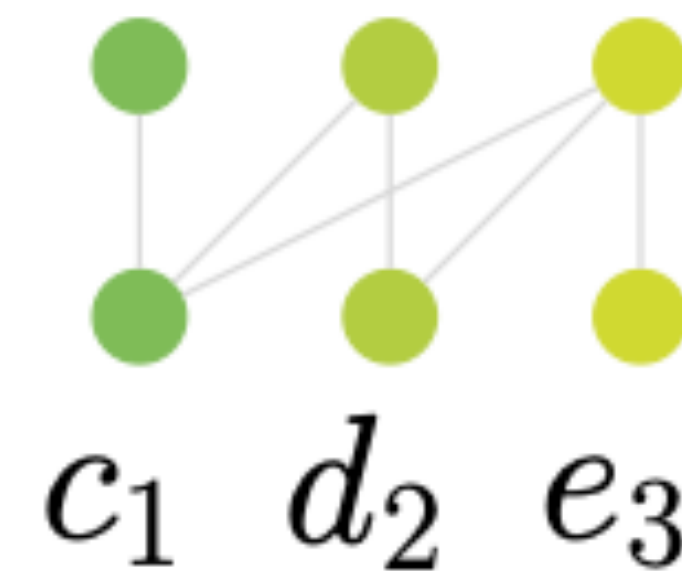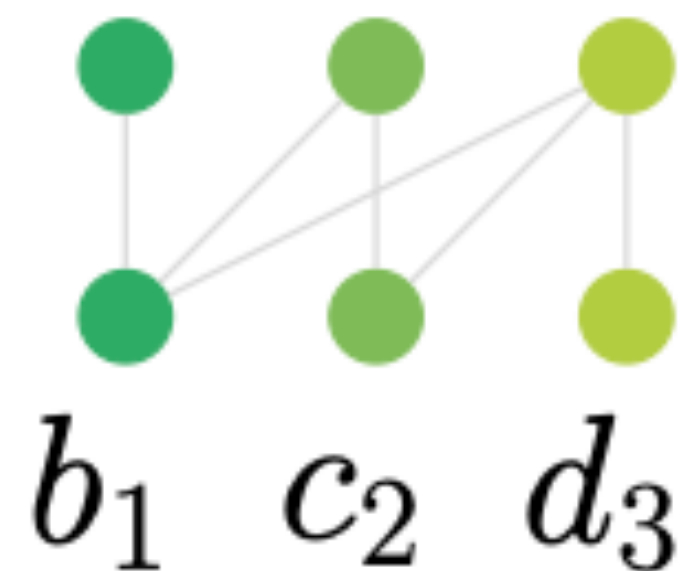
(a) Standard Attention
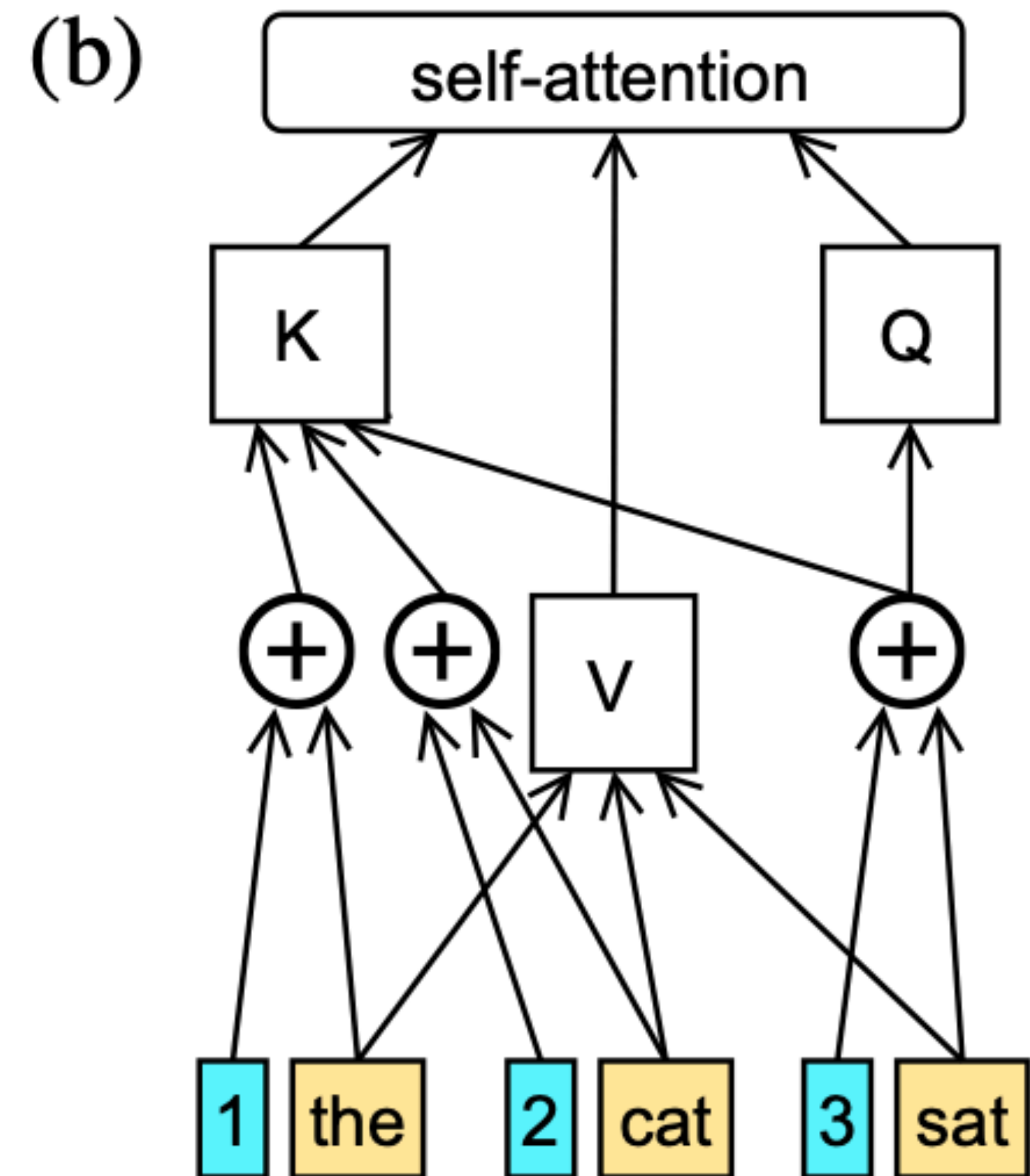
(b) Position Infused Attention (enables caching)

Train on short; inference on long

Non-overlapping

Caching (using PIA)
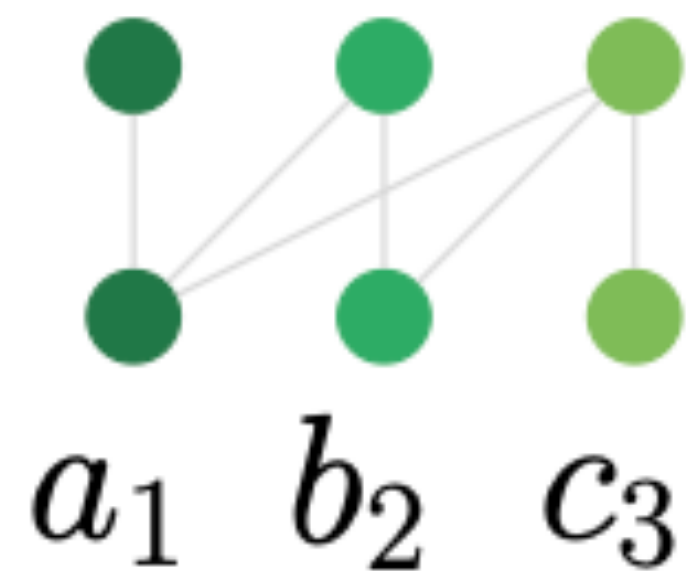
Sliding window with stride S=1

| | Train | Inference | | | |
| | | Nonoverlapping | | Sliding Window (Token-by-token) | |
Subseq. Length	Speed ↑	PPL ↓	Speed ↑	PPL ↓	Speed ↑
32	28.3k	35.37	2.4k	24.98	**74**
64	28.5k	28.03	4.8k	21.47	69
128	**28.9k**	23.81	9.2k	19.76	70
256	28.1k	21.45	14.8k	18.86	63
512	26.1k	20.10	18.1k	18.41	37
1024	22.9k	19.11	**18.3k**	17.97	18
1536	18.4k	19.05	17.1k	18.14	11
3072	13.9k	**18.65**	14.7k	**17.92**	5

First Stage Subseq. Length	Train	Inference
	Speed ↑	PPL ↓
32	21.6k	17.66
64	22.6k	17.56
128	**22.9k**	**17.47**
256	22.5k	17.50
PIA + Cache w/o Staged Training	21.5k	17.85

Model	Param. ↓	Train Speed ↑	Mode	Inference (Test) Speed ↑	PPL ↓
Baseline	**247M**	13.9k	N.o.	**14.7k**	19.40
			S.W.	2.5k	18.70
TransformerXL*	257M	6.0k	N.o.	3.2k	18.30
Sandwich T.	**247M**	13.9k	S.W.	2.5k	17.96
Compressive T.	329M	-	N.o.	-	17.1
Routing T.	-	-	N.o	-	15.8
kNN-LM**	**247M**	13.9k	S.W.	145	**15.79**
PIA + Caching	**247M**	21.5k	N.o.	14.5k	18.55
Staged Training	**247M**	17.6k	S.W.	2.5k	17.56
Shortformer	**247M**	**22.9k**	N.o.	14.5k	18.15

Do We Need to Create Big Datasets to Learn a Task?

Swaroop Mishra* Bhavdeep Sachdeva*
Department of Computer Science, Arizona State University
{srmishr1, bssachde}@asu.edu

https://aclanthology.org/2020.sustainlp-1.23/
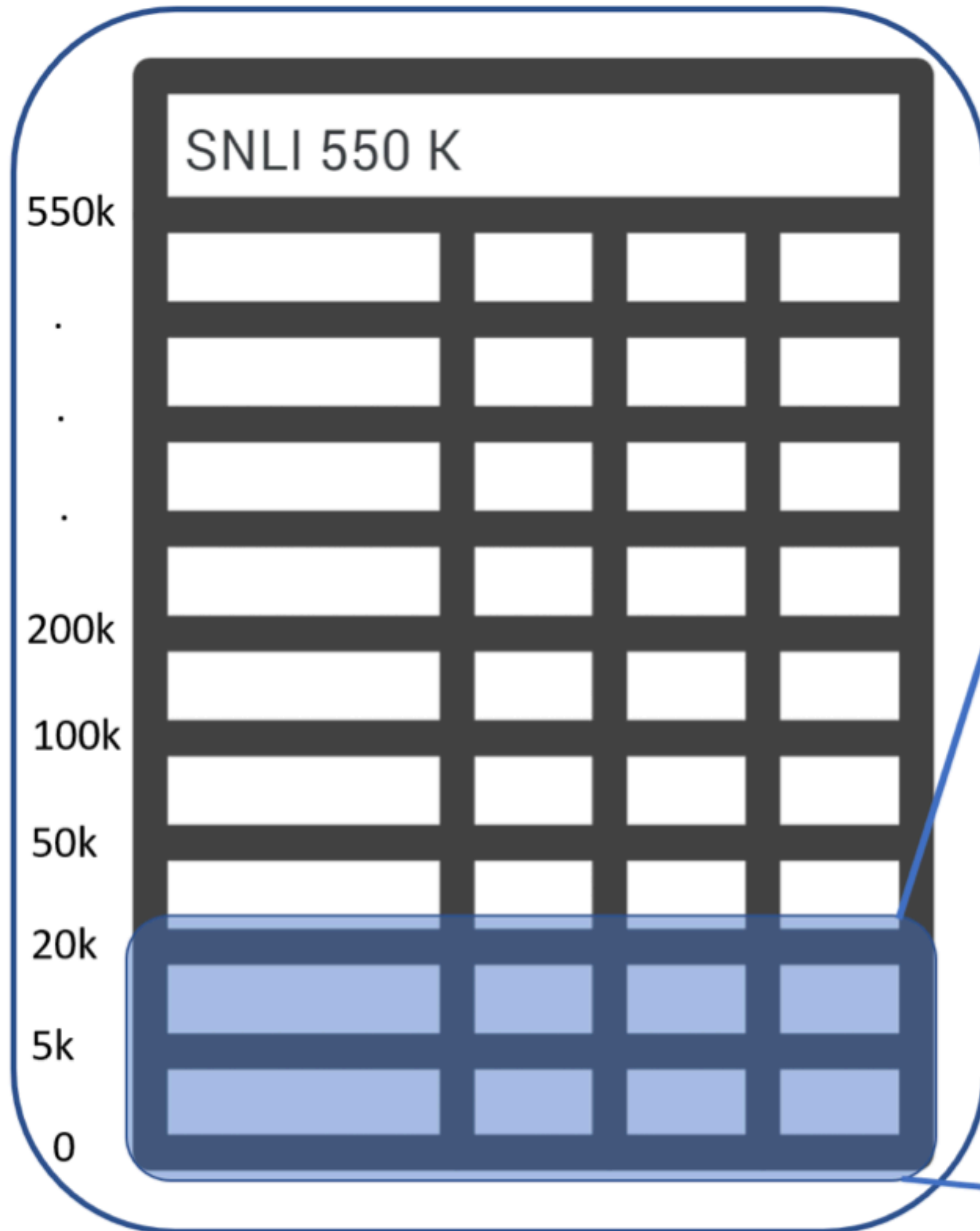
Active Learning for Fine-tuning LLMs

- **Coarse action**

  - Start with random a% of the data and calculate accuracy on heldout data

  - Pick the best performing data and add b% and redo accuracy

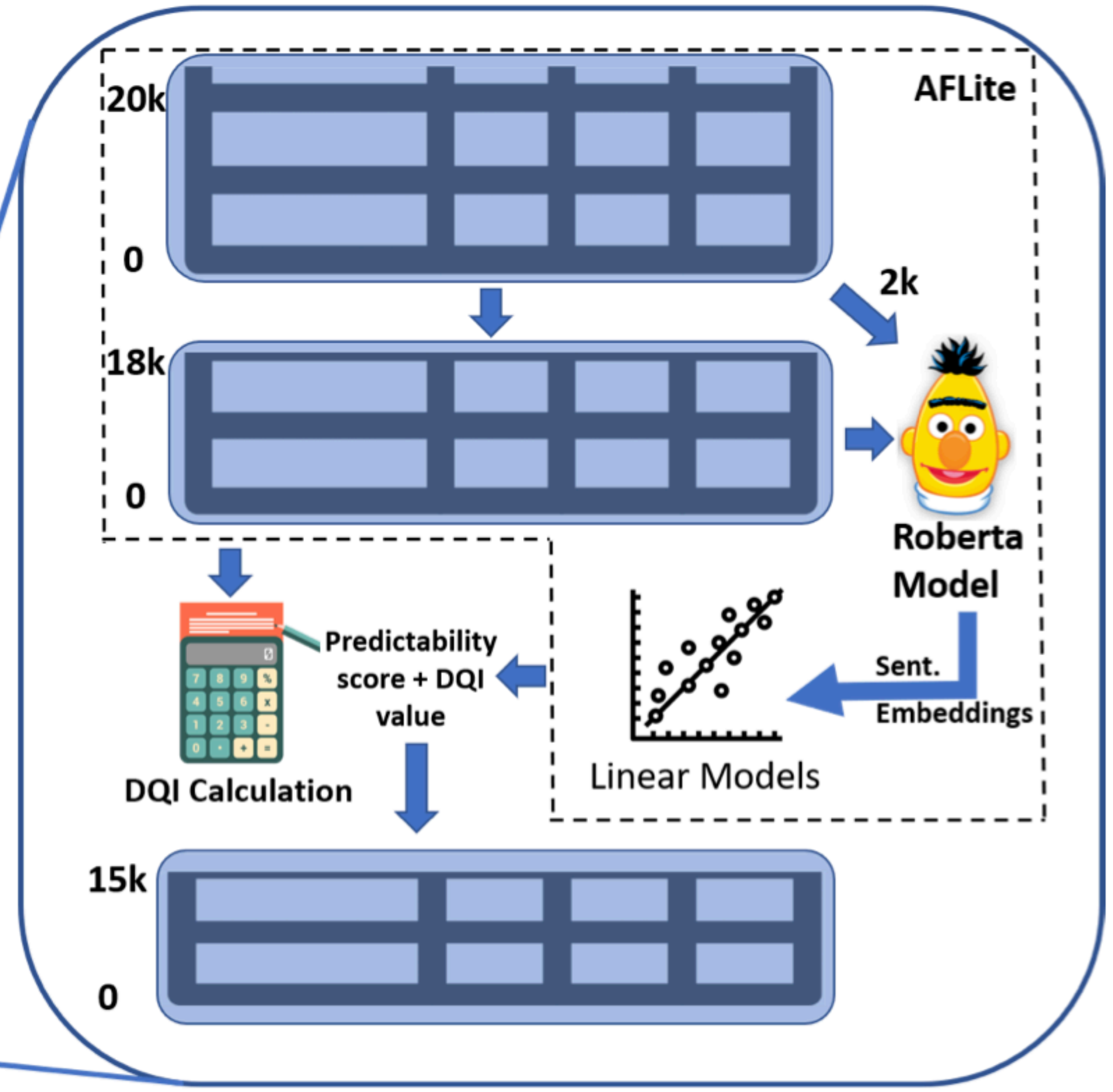 - Continue adding b% of training data until accuracy on heldout data does not increase

- **Fine action**

  - For each training set, randomly drop 10%, randomly divide into train/test, sort by the Data Quality Index (DQI) for bias detection

  - Short list the training data that scores greater than a threshold in DQI

Coarse Action

Fine Action

SNLI 550 K

550k
.
.
.
200k
100k
50k
20k
5k
0

AFLite

20k
0

2k

18k
0

Roberta Model

DQI Calculation

Predictability score + DQI value

Sent. Embeddings

Linear Models

15k
0

Active Learning for Fine-tuning LLMs

- SNLI dataset

- a=5000; b=5000

- **Coarse action** finds 20K training examples

- **Fine action** can further reduce this to 5K-15K

Size	IID Test
550k	**89.64**
20k	84.69
5k	87.47
8k	87.54
10k	87.93
12k	88.56
15k	88.95

Fine

Coarse

Size	Performance on IID test set
5000	36.77
10000	77.45
15000	81.69
20000	**84.69**
25000	80.96