Sparse Distillation: Speeding Up Text Classification by Using Bigger Student Models


Qinyuan Ye[1,†]


Madian Khabsa[2]


Mike Lewis[2]


Sinong Wang[2]


Xiang Ren[1]


Aaron Jaech[2]

† Work mainly done while interning at Meta AI

1 USC Viterbi Department of Computer Science

2 Meta AI

Motivation

Want a faster model for your NLP task?

Your go-to method

distill

RoBERTa-Large
24-layer, 1024-hidden
16-heads, 355M parameters

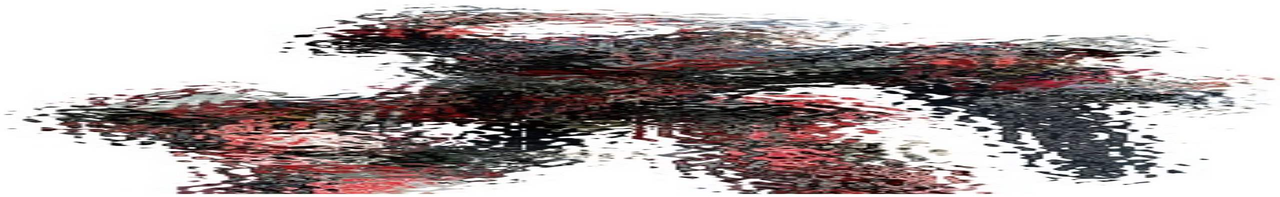
Distill-RoBERTa
6-layer, 768-hidden
12-heads, 82M parameters

Motivation

Want a faster model for your NLP task?

What if ?...

distill



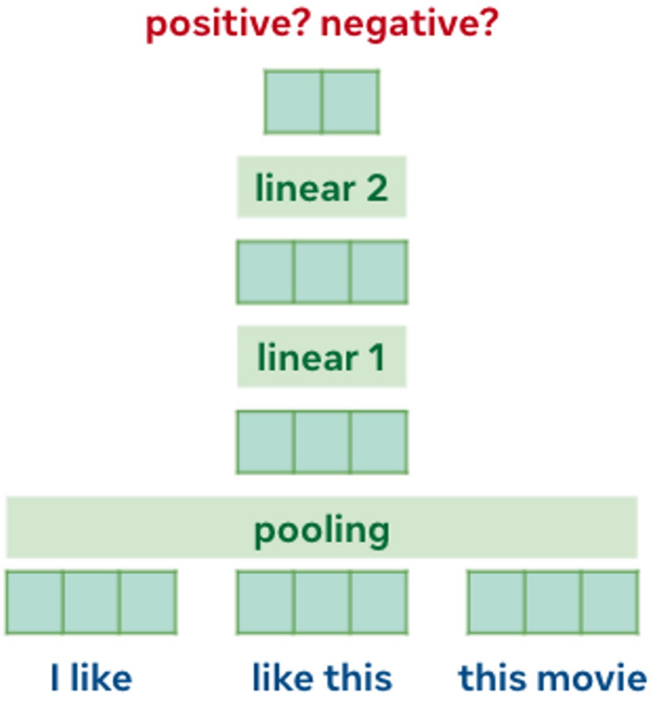
Something big, sparse, shallow, and fast!

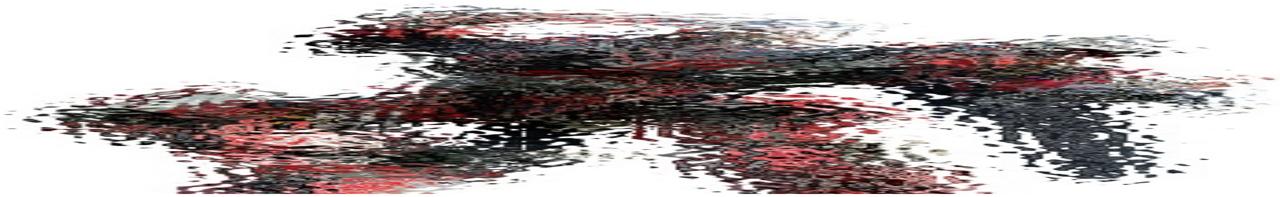
RoBERTa-Large
24-layer, 1024-hidden
16-heads, 355M parameters

Motivation

Deep Averaging Networks (DANs)
(Iyyer et al., 2015)

In 2015, 2-layer MLPs are considered "deep" networks

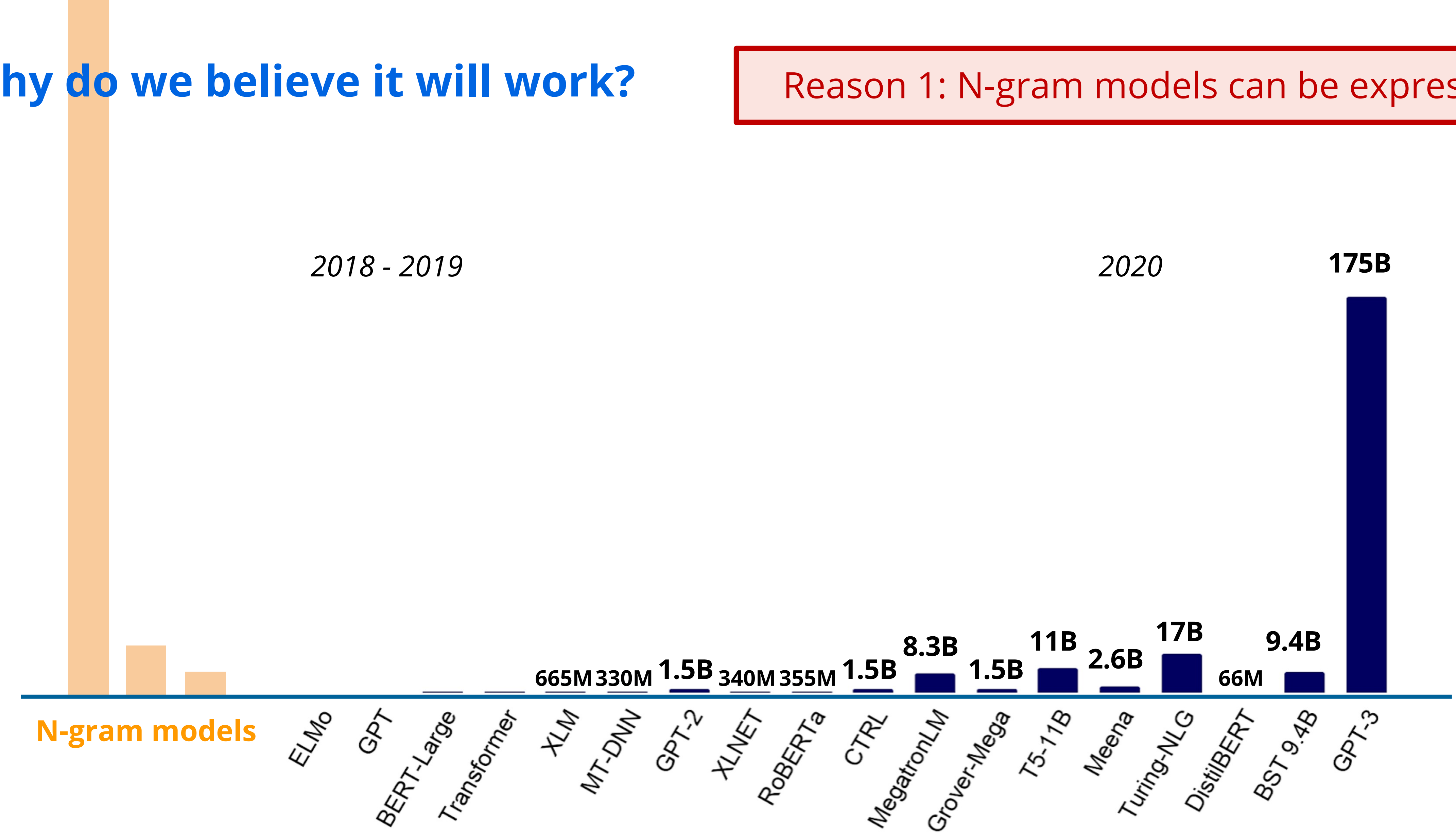




Something big, sparse, shallow, and fast!

Why do we believe it will work?

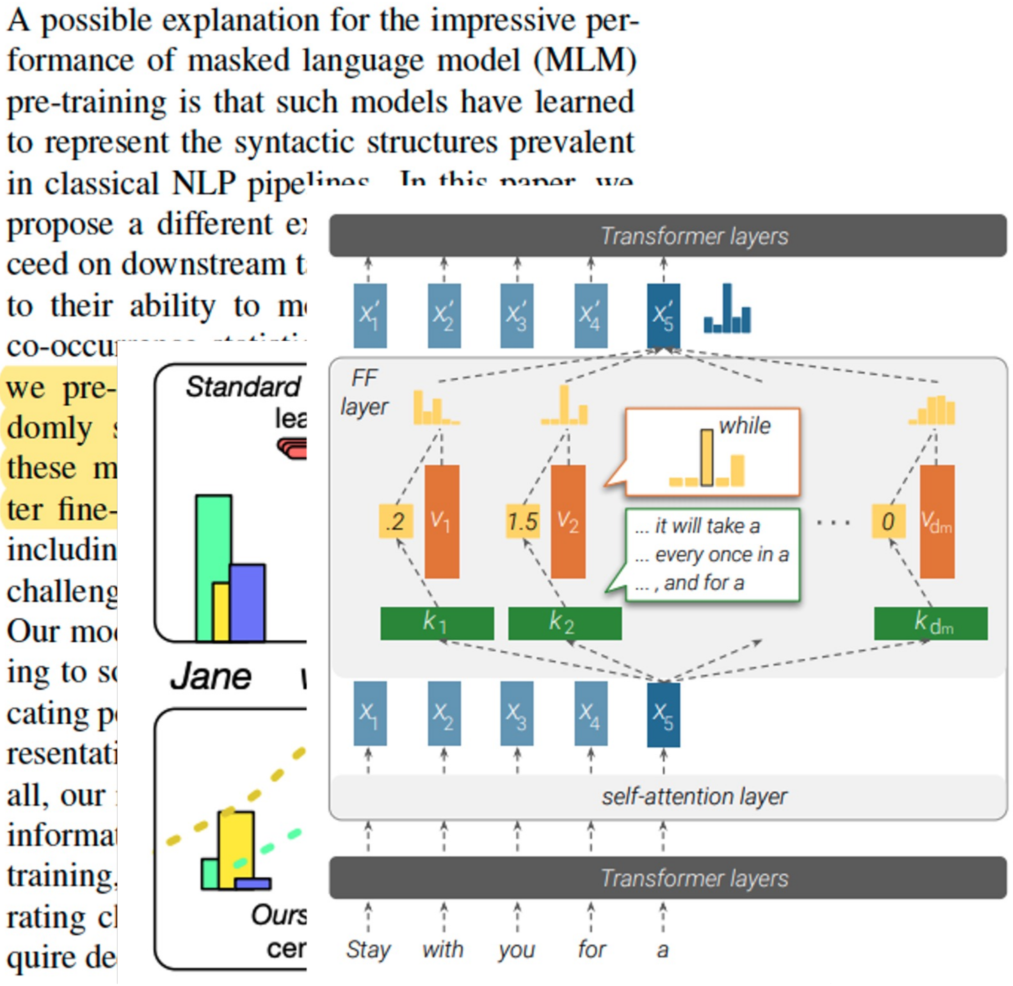
Reason 1: N-gram models can be expressive!



Depending on n-gram selection,
the model can have billions of parameters!

Why do we believe it will work?

Reason 2: Some expensive operations in transformers may not be necessary



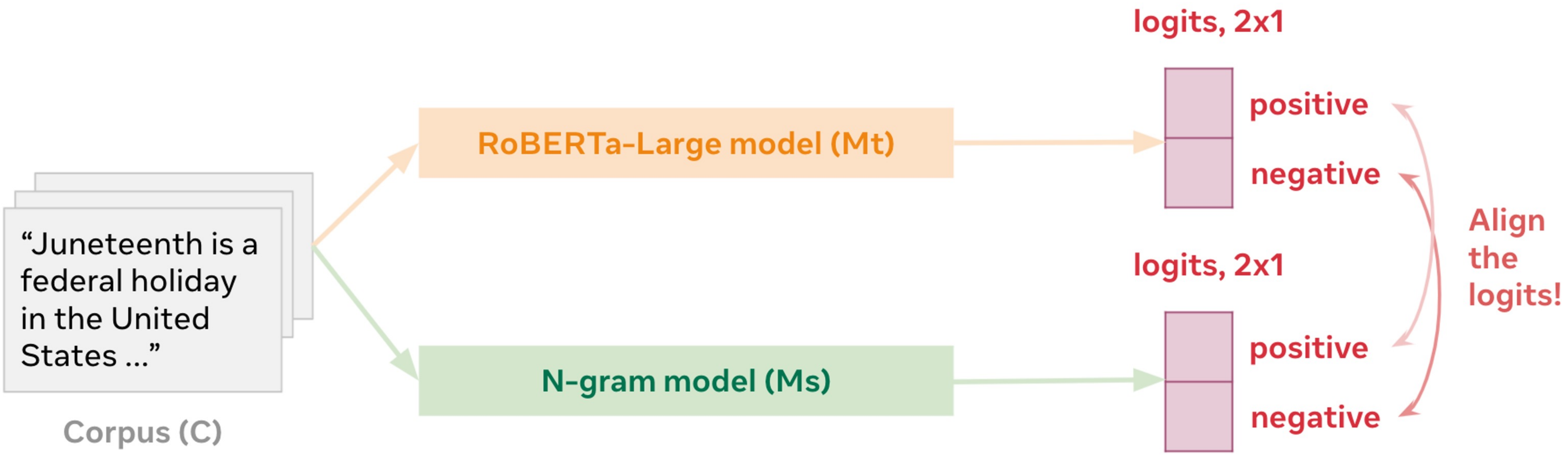
Word order does not matter?

Local attention is good enough?

Still memorizing patterns?

Sparse Distillation

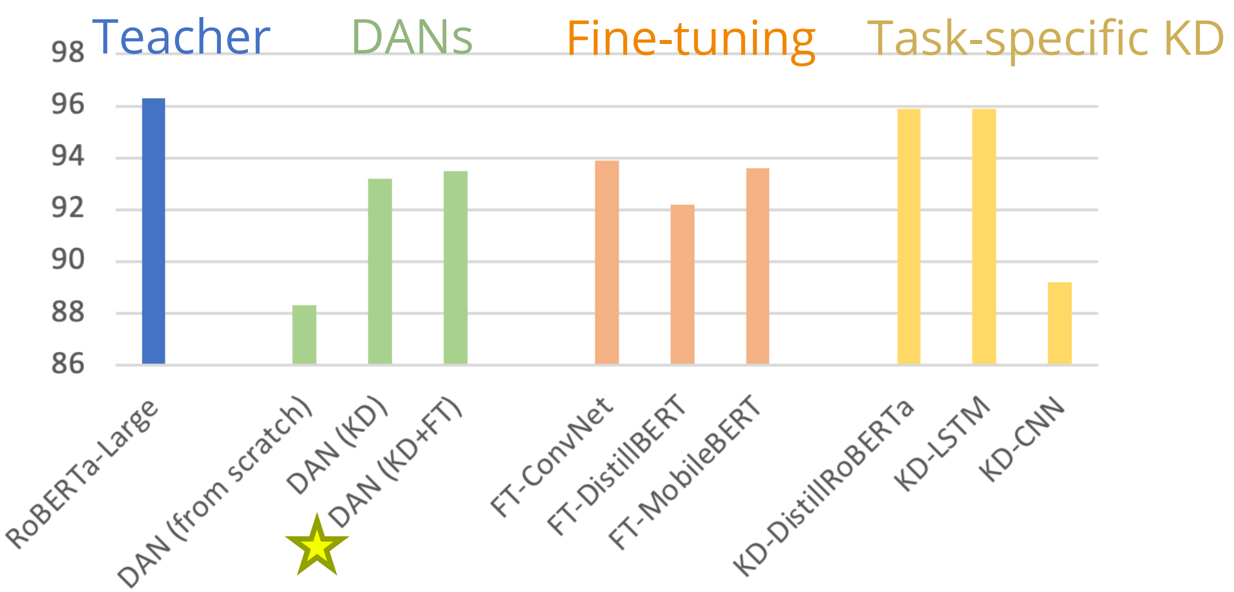
- Given a text classification task (T), we ...
    - 1. Fine-tune a RoBERTa-Large model and use it as the teacher model **(Mt)**
    - 2. Apply the teacher model to some in-domain corpus **(C)**, save the logits.
    - 3. Use knowledge distillation and the saved logits to train the student model **(Ms)**



Finding 1: 600x speed up, <3% performance drop

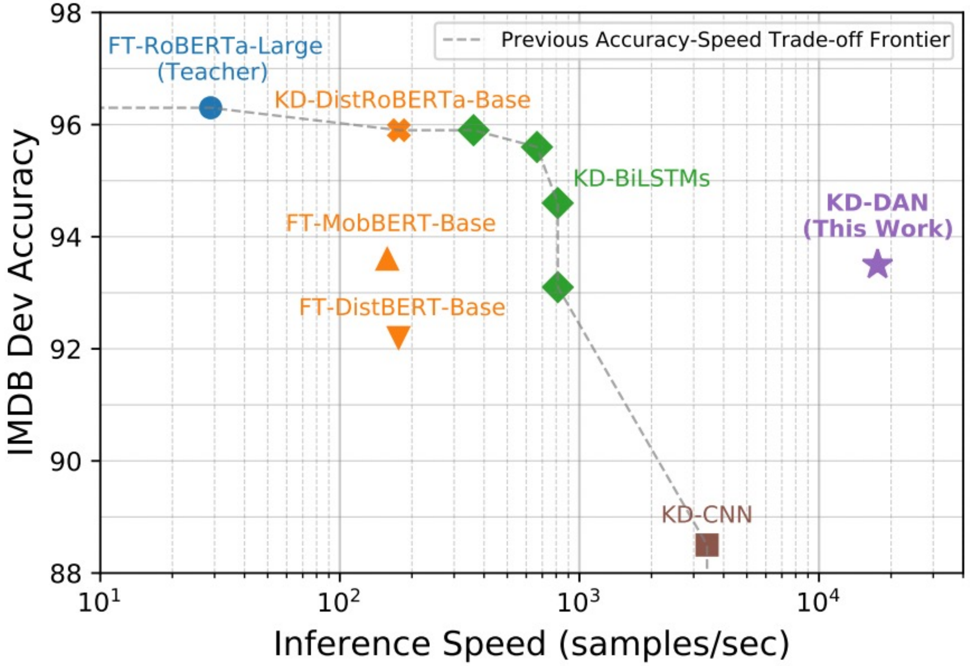
- Take IMDB review classification as an example

IMDB Dev Performance



DAN (KD+FT) can *match* **fine-tuning** baselines
Within 3% gap compared to other methods

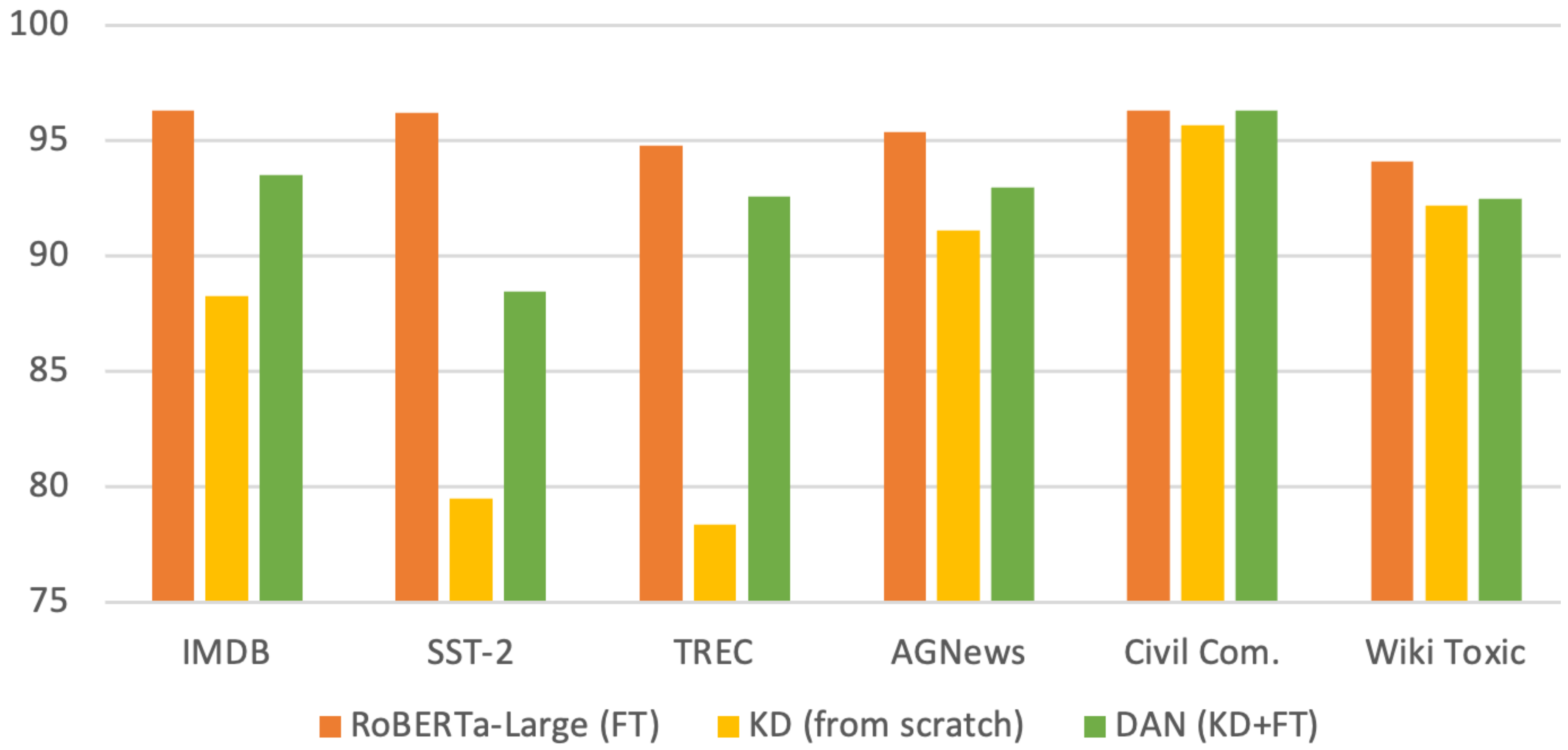
Performance vs. Inference Speed



***600x faster* than other methods**

Finding 1: 600x speed up, <3% performance drop

- Experimenting with Sparse Distillation on **6 single-sentence classification tasks**



KD always brings improvement

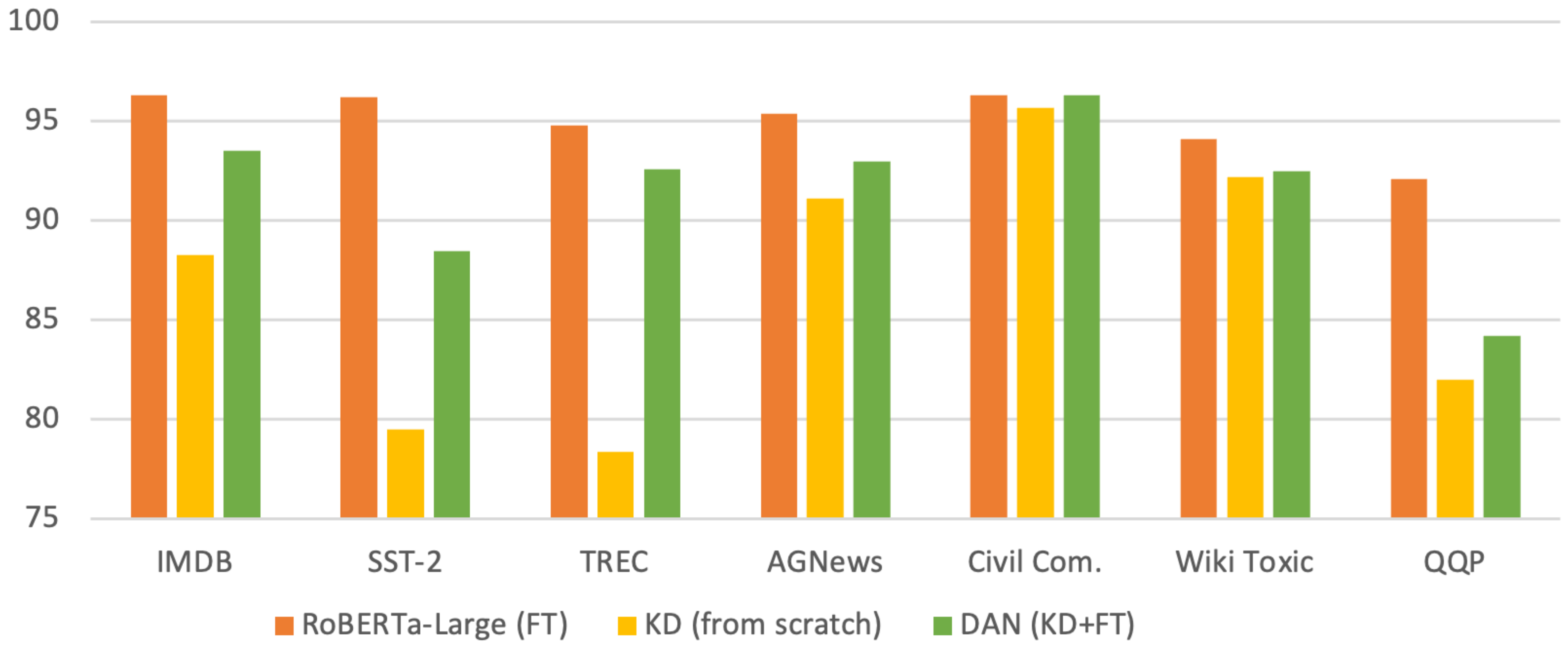
On average, DAN (KD+FT) maintains 97% of the teacher's performance

Finding 1: 600x speed up, <3% performance drop

- Extending Sparse Distillation to **a sentence-pair task**

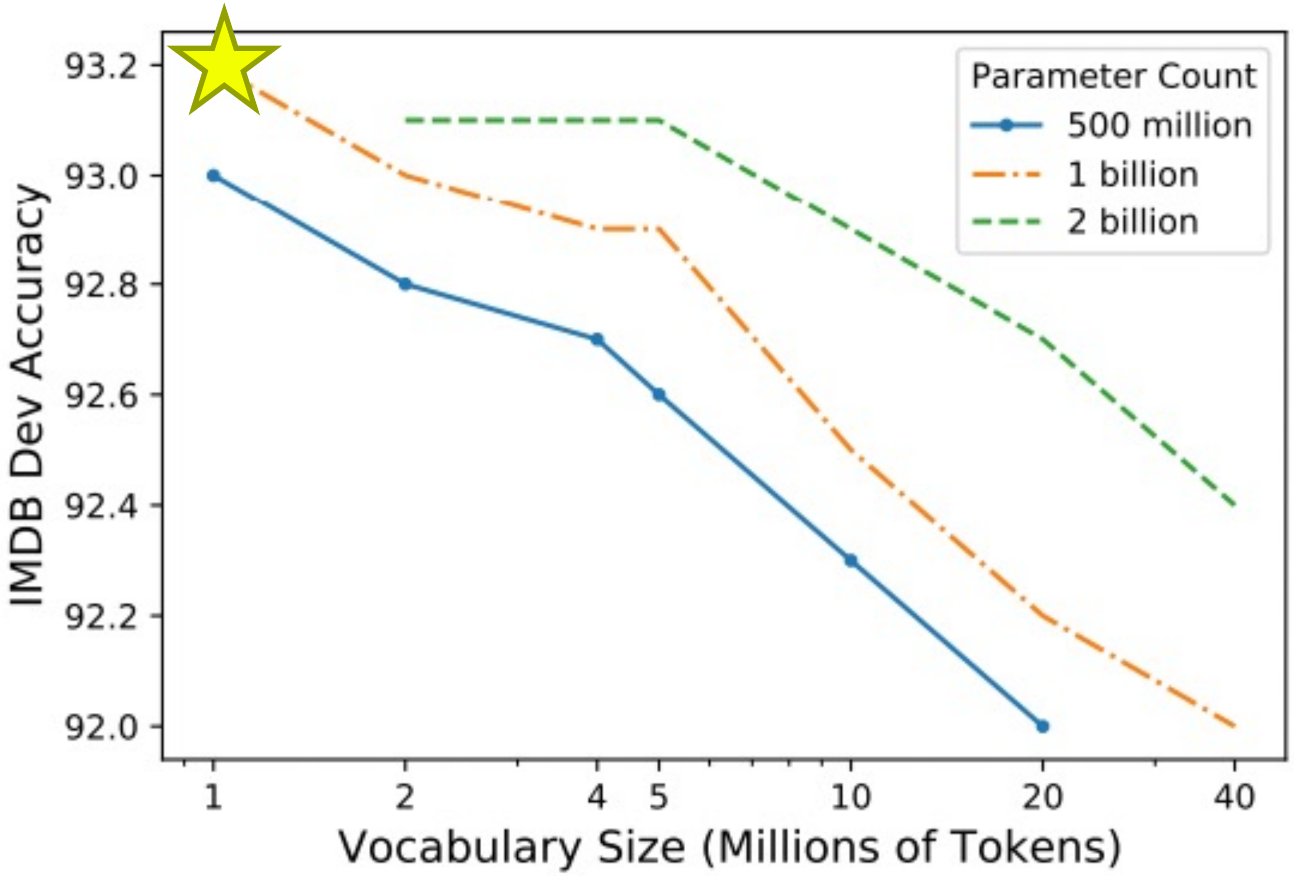
KD still brings improvement

Gap is bigger on sentence-pair tasks



Finding 2: Use smaller vocab and large embedding dimension

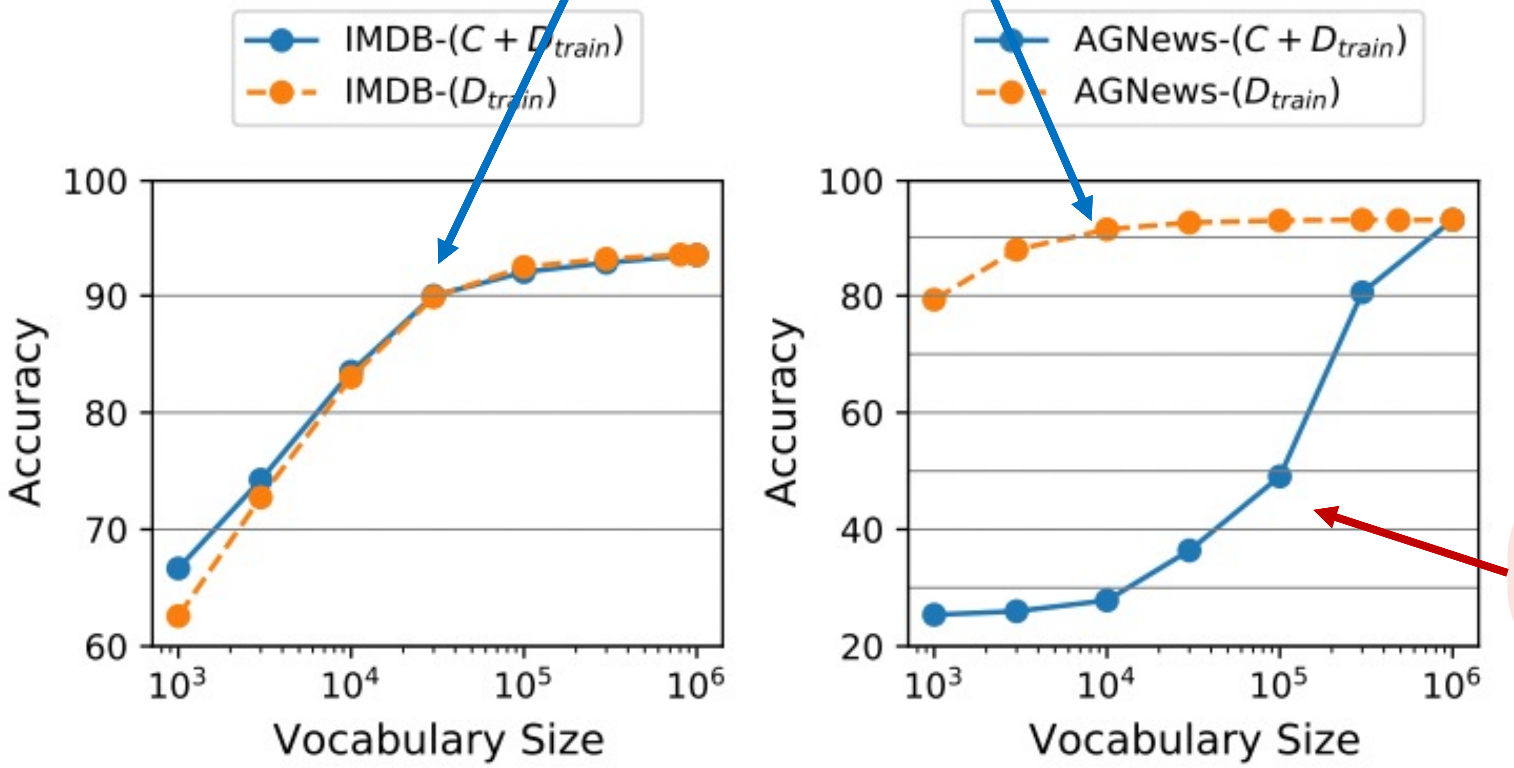
- We different parameter budgets (**500 millions**, **1 billion**, **2 billions**)



Finding 3: Can flexibly prune the model

- Pruning the model's least frequent n-grams …

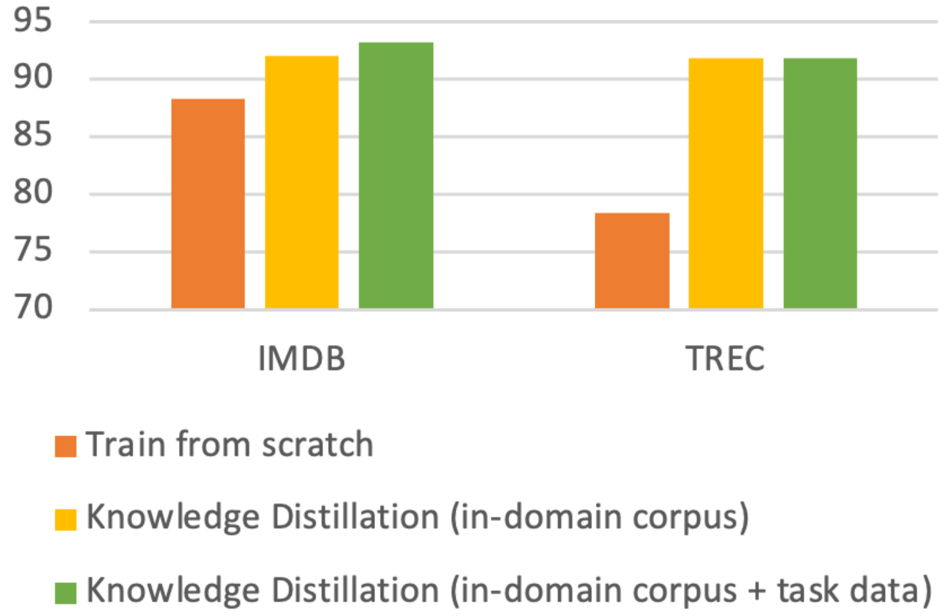
Decent accuracy (90%) when model is cut to 3% of its original size



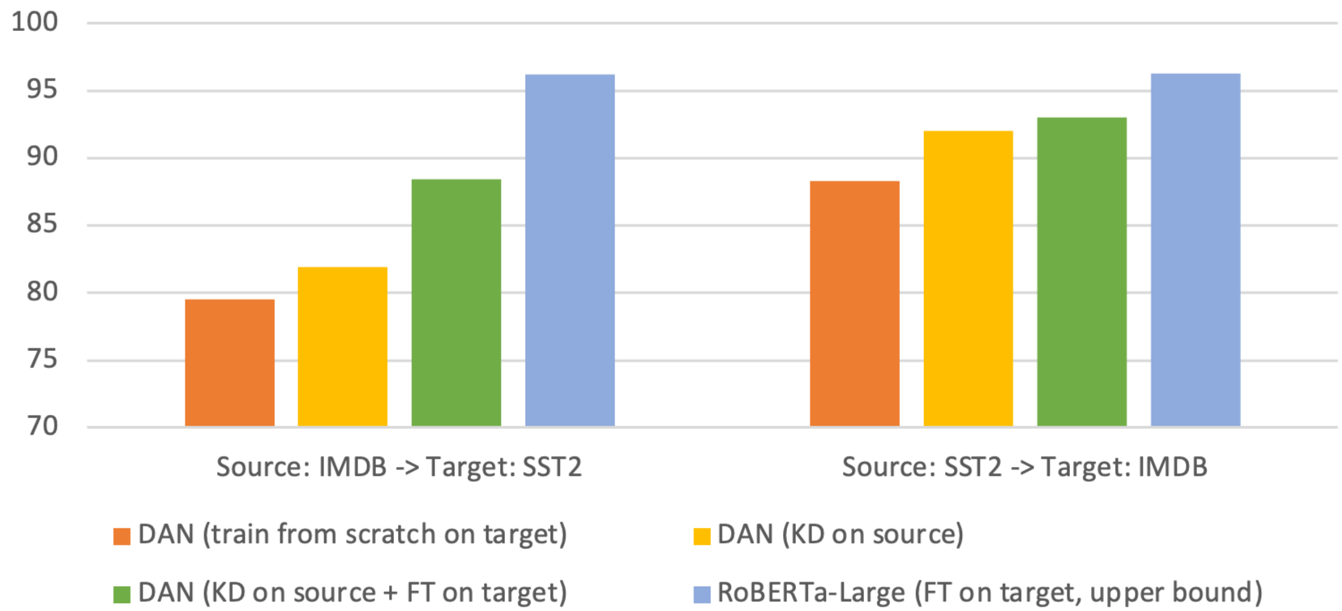
But be careful with how the frequency is computed!

Finding 4: Beneficial in various practical settings

Privacy Preserving Setting



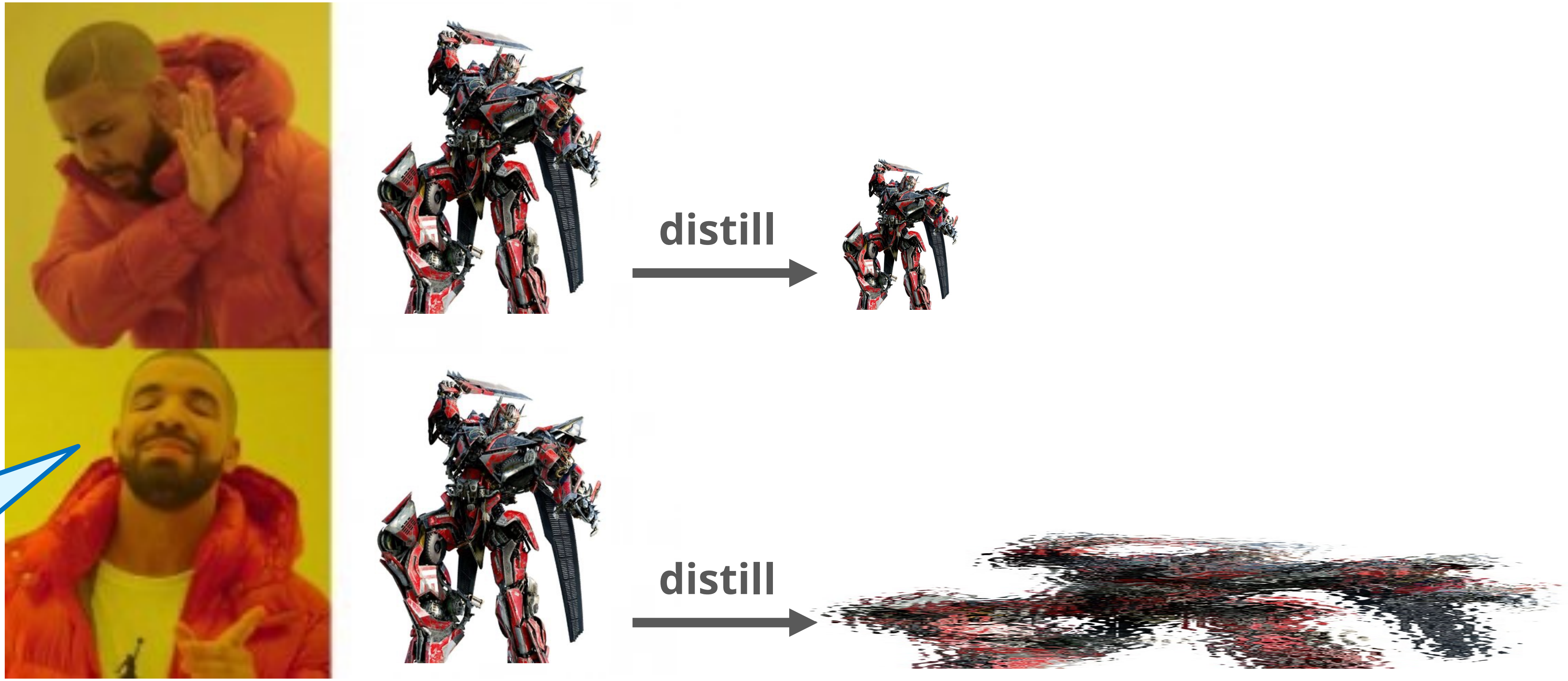
Domain Adaptation / Generalization Setting



Conclusions

- We introduce **Sparse Distillation**, a framework that distills transformers into models that maintain ***competitive performance***, while achieving ***up to 600x speed up***.
- Counter-intuitively, the student model we use has ***more parameters*** than the teacher model -- The student model aggressively cuts off computation cost by compensating it with more parameters.
- Sparse Distillation is useful in many ***practical*** scenarios: flexible post-hoc pruning, helpful in privacy-preserving setting, helpful in domain generalization / adaptation setting.

Conclusions (in a Meme 🤣)



3x bigger
sparsely-activated
600x faster
strong performance