BERT (Part 2)

CS 685, Spring 2021

Advanced Topics in Natural Language Processing

http://brenocon.com/cs685

https://people.cs.umass.edu/~brenocon/cs685_s21/

Brendan O'Connor

College of Information and Computer Sciences

University of Massachusetts Amherst

- OK this BERT sounds nice, but
  - What does it learn? How do we use it?
    Rogers et al. (2020), TACL
  - What about those important-seeming details?
    Liu et al. (2019), RoBERTa, arxiv
  - Does the multilingual training really work?
    Pires et al. (2019), ACL
  - Do you need all the 16 heads?
    Michel et al. (2019), NeurIPS
  - How reliable is fine-tuning?
    Dodge et al. (2020), arxiv

A Primer in BERTology: What We Know About How BERT Works

Anna Rogers
Center for Social Data Science
University of Copenhagen
arogers@sodas.ku.dk

Olga Kovaleva
Dept. of Computer Science
University of
Massachusetts Lowell
okovalev@cs.uml.edu

Anna Rumshisky
Dept. of Computer Science
University of
Massachusetts Lowell
arum@cs.uml.edu

Abstract

Transformer-based models have pushed state of the art in many areas of NLP, but our understanding of what is behind their success is still limited. This paper is the first survey of over 150 studies of the popular BERT model. We review the current state of knowledge about how BERT works, what kind of information it learns and how it is represented, common modifications to its training objectives and architecture, the overparameterization issue, and approaches to compression. We then outline directions for future research.

- Types of knowledge in or not in BERT
  - Syntactic
  - Semantic
  - World knowledge
- *Where* in BERT (layers? self-attn heads?) is this info?
- Overview of fine-tuning and model compression methods

What do heads learn??

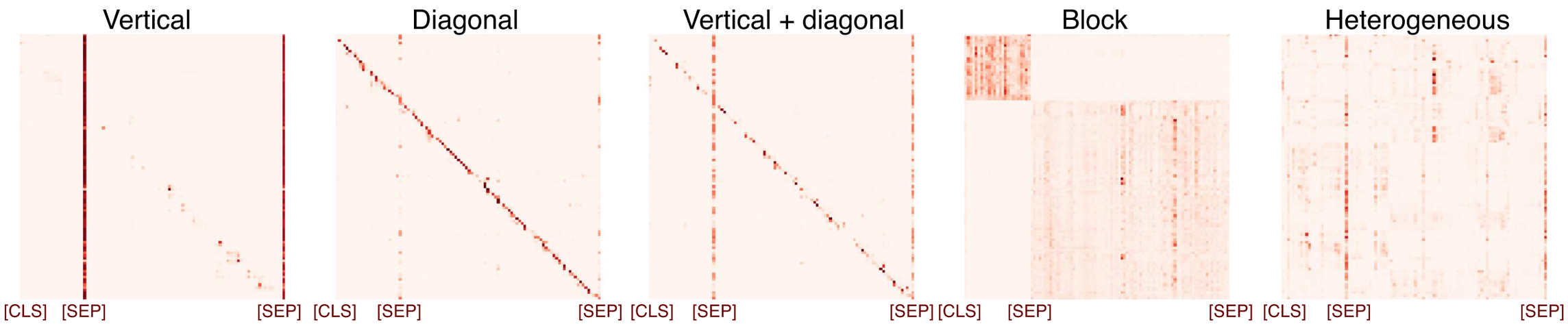


Figure 3: Attention patterns in BERT (Kovaleva et al., 2019).

Are Sixteen Heads Really Better than One?

Paul Michel
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA
pmichel1@cs.cmu.edu

Omer Levy
Facebook Artificial Intelligence Research
Seattle, WA
omerlevy@fb.com

Graham Neubig
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA
gneubig@cs.cmu.edu

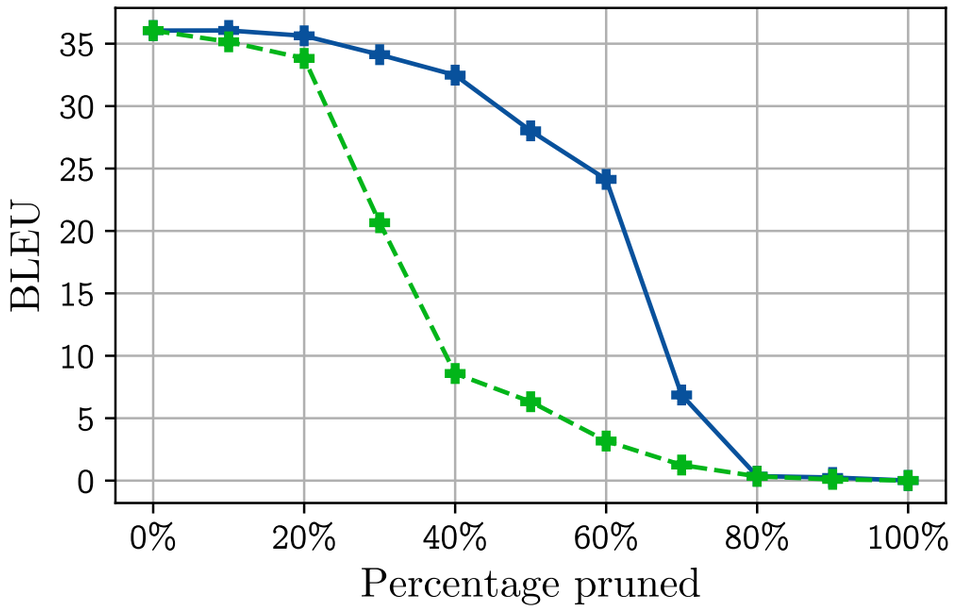
4.1 Head Importance Score for Pruning

As a proxy score for head importance, we look at the expected sensitivity of the model to the mask variables ξ_h defined in §2.3:

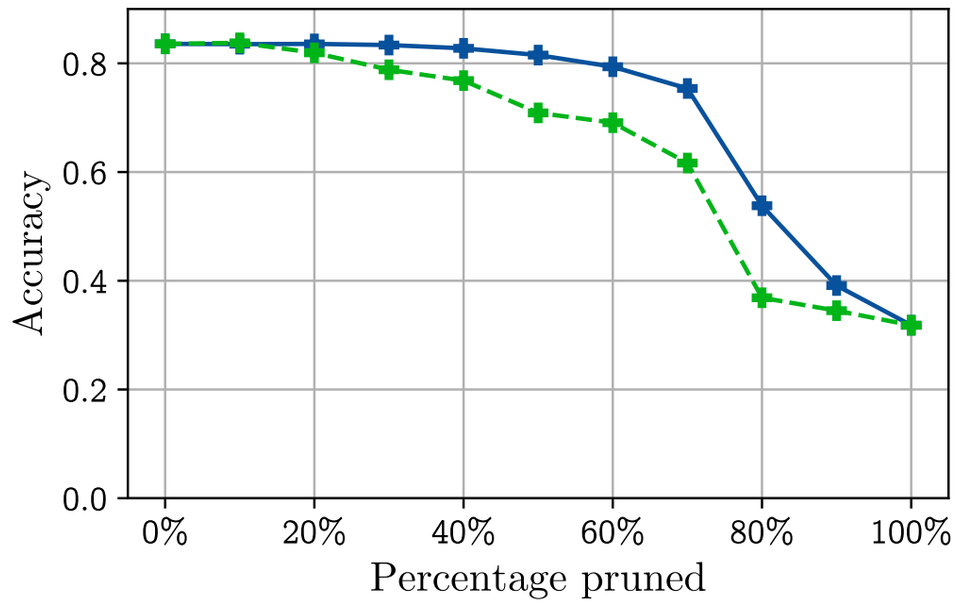
$$I_h = \mathbb{E}_{x \sim X} \left| \frac{\partial \mathcal{L}(x)}{\partial \xi_h} \right| \tag{2}$$

Are Sixteen Heads Really Better than One?

- **Not "no," but…**



(a) Evolution of BLEU score on `newstest2013` when heads are pruned from WMT.



(b) Evolution of accuracy on the MultiNLI-matched validation set when heads are pruned from BERT.

Figure 3: Evolution of accuracy by number of heads pruned according to I_h (solid blue) and individual oracle performance difference (dashed green).

On the other hand

- Several studies find suggestions that heads, or combinations of them, may specialize in syntactic relations (at least, more so than chance)
- Are these findings consistent?

RoBERTa: A Robustly Optimized BERT Pretraining Approach

Yinhan Liu[*§] **Myle Ott**[*§] **Naman Goyal**[*§] **Jingfei Du**[*§] **Mandar Joshi**[†]
Danqi Chen[§] **Omer Levy**[§] **Mike Lewis**[§] **Luke Zettlemoyer**[†§] **Veselin Stoyanov**[§]

[†] Paul G. Allen School of Computer Science & Engineering,
University of Washington, Seattle, WA
{mandar90,lsz}@cs.washington.edu

[§] Facebook AI
{yinhanliu,myleott,naman,jingfeidu,
danqi,omerlevy,mikelewis,lsz,ves}@fb.com

Abstract

Language model pretraining has led to significant performance gains but careful comparison between different approaches is challenging. Training is computationally expensive, often done on private datasets of different sizes, and, as we will show, hyperparameter choices have significant impact on the final results. We present a replication study of BERT pretraining (Devlin et al., 2019) that carefully measures the impact of many key hyperparameters and training data size. We find that BERT was significantly undertrained, and can match or exceed the performance of every model published after it. Our best model achieves state-of-the-art results on GLUE, RACE and SQuAD. These results highlight the importance of previously overlooked design choices,

We present a replication study of BERT pretraining (Devlin et al., 2019), which includes a careful evaluation of the effects of hyperparmeter tuning and training set size. We find that BERT was significantly undertrained and propose an improved recipe for training BERT models, which we call RoBERTa, that can match or exceed the performance of all of the post-BERT methods. Our modifications are simple, they include: (1) training the model longer, with bigger batches, over more data; (2) removing the next sentence prediction objective; (3) training on longer sequences; and (4) dynamically changing the masking pattern applied to the training data. We also collect a large new dataset (CC-News) of comparable size to other privately used datasets, to better control for training set size effects.

Model	SQuAD 1.1/2.0	MNLI-m	SST-2	RACE
Our reimplementation (with NSP loss):				
SEGMENT-PAIR	90.4/78.7	84.0	92.9	64.2
SENTENCE-PAIR	88.7/76.2	82.9	92.1	63.0
Our reimplementation (without NSP loss):				
FULL-SENTENCES	90.4/79.1	84.7	92.5	64.8
DOC-SENTENCES	90.6/79.7	84.7	92.7	65.6
$\text{BERT}_{\text{BASE}}$	88.5/76.3	84.3	92.8	64.3
$\text{XLNet}_{\text{BASE}}$ (K = 7)	–/81.3	85.8	92.7	66.1
$\text{XLNet}_{\text{BASE}}$ (K = 6)	–/81.0	85.6	93.4	66.7

Table 2: Development set results for base models pretrained over BOOKCORPUS and WIKIPEDIA. All models are trained for 1M steps with a batch size of 256 sequences. We report F1 for SQuAD and accuracy for MNLI-m, SST-2 and RACE. Reported results are medians over five random initializations (seeds). Results for $\text{BERT}_{\text{BASE}}$ and $\text{XLNet}_{\text{BASE}}$ are from Yang et al. (2019).

- Next sentence prediction: oops doesn't matter

Model	data	bsz	steps	SQuAD (v1.1/2.0)	MNLI-m	SST-2
RoBERTa						
with BOOKS + WIKI	16GB	8K	100K	93.6/87.3	89.0	95.3
+ additional data (§3.2)	160GB	8K	100K	94.0/87.7	89.3	95.6
+ pretrain longer	160GB	8K	300K	94.4/88.7	90.0	96.1
+ pretrain even longer	160GB	8K	500K	**94.6/89.4**	**90.2**	**96.4**
$\text{BERT}_{\text{LARGE}}$						
with BOOKS + WIKI	13GB	256	1M	90.9/81.8	86.6	93.7
$\text{XLNet}_{\text{LARGE}}$						
with BOOKS + WIKI	13GB	256	1M	94.0/87.8	88.4	94.4
+ additional data	126GB	2K	500K	94.5/88.8	89.8	95.6

Table 4: Development set results for RoBERTa as we pretrain over more data (16GB $\rightarrow$ 160GB of text) and pretrain for longer (100K $\rightarrow$ 300K $\rightarrow$ 500K steps). Each row accumulates improvements from the rows above. RoBERTa

- How much does the model matter, versus more data and more training?

Multilingual BERT

google-research / **bert**

Watch 918 | Star 21.7k

Code | Issues 574 | Pull requests 63 | Actions | Projects 0 | Wiki | Security | Insights

Docs: Very wrong assertion that Wikipedia size correlates with number of speakers #760

Edit

Closed **brendano** opened this issue on Jul 12, 2019 · 3 comments

brendano commented on Jul 12, 2019 • edited

Currently `bert/multilingual.md` states:

"... the size of the Wikipedia for a given language varies greatly, and therefore low-resource languages may be "under-represented" in terms of the neural network model ...
However, the size of a Wikipedia also correlates with the number of speakers of a language"

Assignees

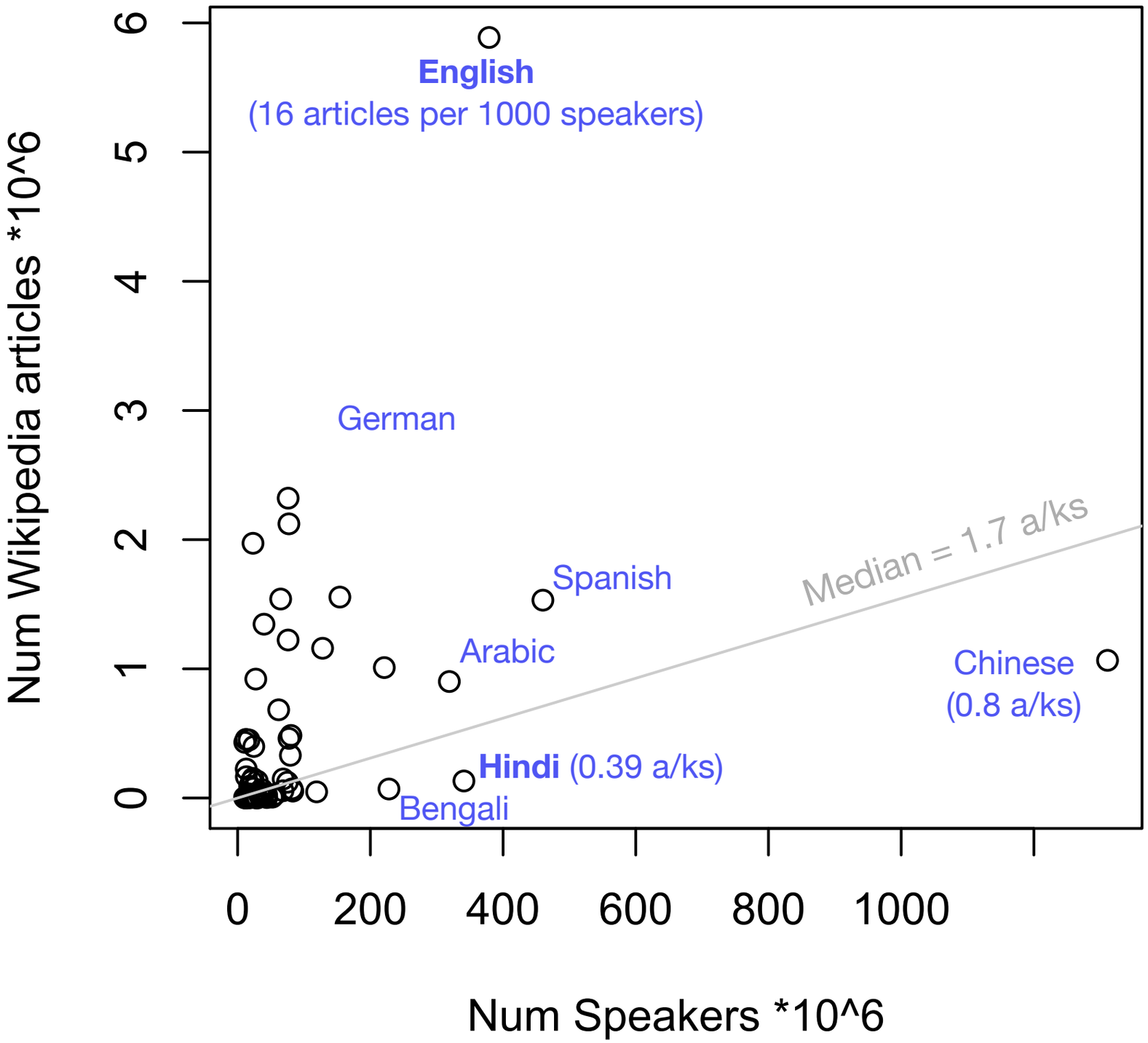
slavpetrov

Labels

None yet

Disparity in language resources

Will pretraining on large unlabeled corpora solve NLU?



How multilingual is Multilingual BERT?

Telmo Pires* **Eva Schlinger** **Dan Garrette**
Google Research
{telmop,eschling,dhgarrette}@google.com

Our results show that M-BERT is able to perform cross-lingual generalization surprisingly well. More importantly, we present the results of a number of probing experiments designed to test various hypotheses about how the model is able to perform this transfer. Our experiments show that while high lexical overlap between languages improves transfer, M-BERT is also able to transfer between languages written in different scripts—thus having *zero* lexical overlap—indicating that it captures multilingual representations. We further show that transfer works best for typologically similar languages, suggesting that while M-BERT's multilingual representation is able to map learned structures onto new vocabularies, it does not seem to learn systematic transformations of those structures to accommodate a target language with different word order.

- Zero-shot transfer
  - 1. Fine-tune for task on language X
  - 2. *Evaluate* on language Y

Fine-tuning \ Eval	EN	DE	ES	IT
EN	**96.82**	89.40	85.91	91.60
DE	83.99	**93.99**	86.32	88.39
ES	81.64	88.87	**96.71**	93.71
IT	86.79	87.82	91.28	**98.11**

Table 2: POS accuracy on a subset of UD languages.

	HI	UR
HI	**97.1**	85.9
UR	91.1	**93.8**

	EN	BG	JA
EN	**96.8**	87.1	49.4
BG	82.2	**98.9**	51.6
JA	57.4	67.2	**96.5**

Table 4: POS accuracy on the UD test set for languages with different scripts. Row=fine-tuning, column=eval.

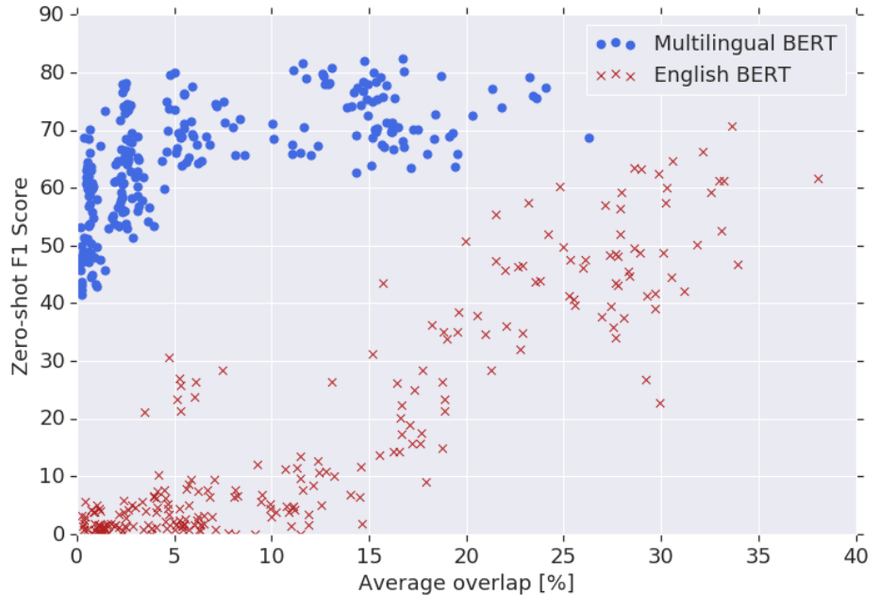


Figure 1: Zero-shot NER F1 score versus entity word piece overlap among 16 languages. While performance using EN-BERT depends directly on word piece overlap, M-BERT's performance is largely independent of overlap, indicating that it learns multilingual representations deeper than simple vocabulary memorization.

- Though results for code-switching and transliterated transfer worse than previous work targeting those problems

Fine-Tuning Pretrained Language Models: Weight Initializations, Data Orders, and Early Stopping

Jesse Dodge [1 2] **Gabriel Ilharco** [3] **Roy Schwartz** [2 3] **Ali Farhadi** [2 3 4] **Hannaneh Hajishirzi** [2 3] **Noah Smith** [2 3]

Abstract

Fine-tuning pretrained contextual word embedding models to supervised downstream tasks has become commonplace in natural language processing. This process, however, is often brittle: even with the same hyperparameter values, distinct random seeds can lead to substantially different results. To better understand this phenomenon, we experiment with four datasets from the GLUE benchmark, fine-tuning BERT hundreds of times on each while varying only the random seeds. We

On small datasets, we observe that many fine-tuning trials diverge part of the way through training, and we offer best practices for practitioners to stop training less promising runs early. We

	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

Table 1. Fine-tuning BERT multiple times while varying only random seeds leads to substantial improvements over previously published validation results with the same model and experimental

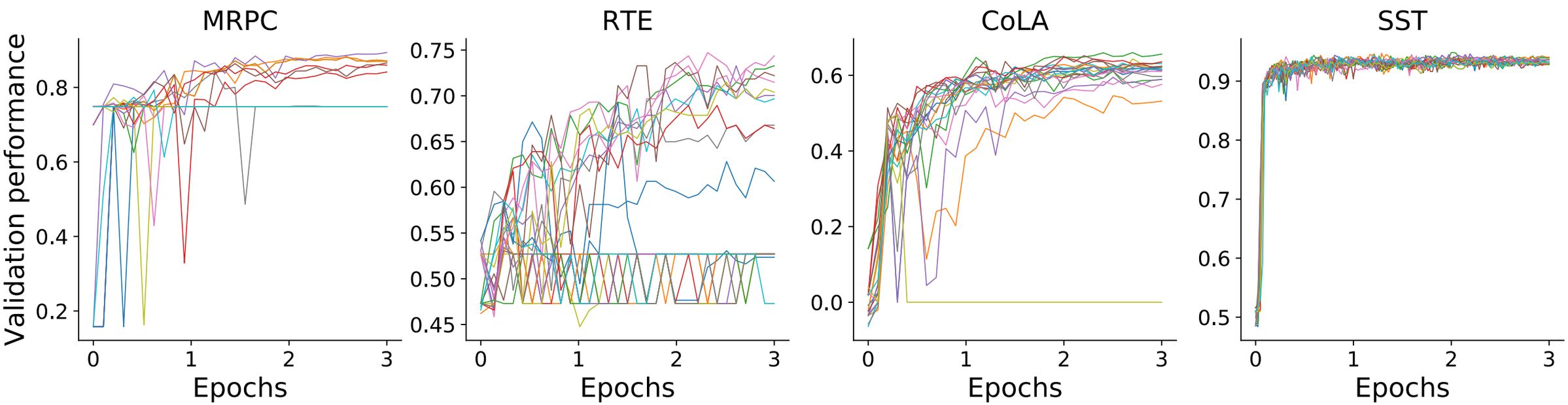


Figure 4. Some promising seeds can be distinguished early in training. The plots show training curves for 20 random WI and DO

- IMO "random seed" really means "random trial"
  A “good random seed” means a useful combination of
  - random parameter initialization
  - random data order

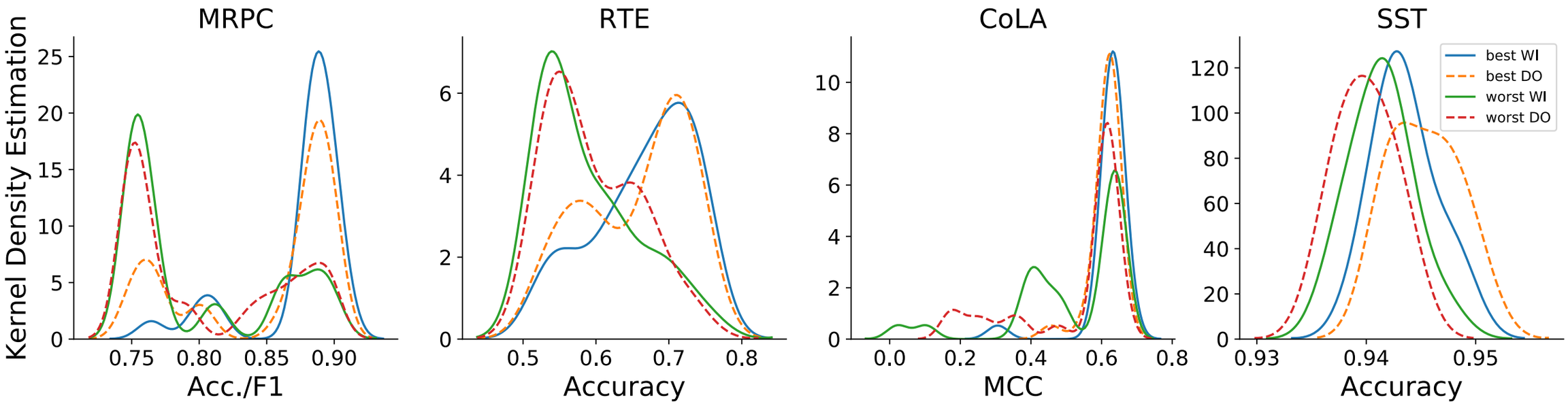


Figure 3. Some seeds are better then others. Plots show the kernel density estimation of the distribution of validation performance for best and worst WI and DO seeds. Curves for DO seeds are shown in dashed lines and for WI in solid lines. MRPC and RTE exhibit pronounced bimodal shapes, where one of the modes represents divergence; models trained with the worst WI and DO are more likely to diverge than learn to predict better than random guessing. Compared to the best seeds, the worst seeds are conspicuously more densely populated in the lower performing regions, for all datasets.

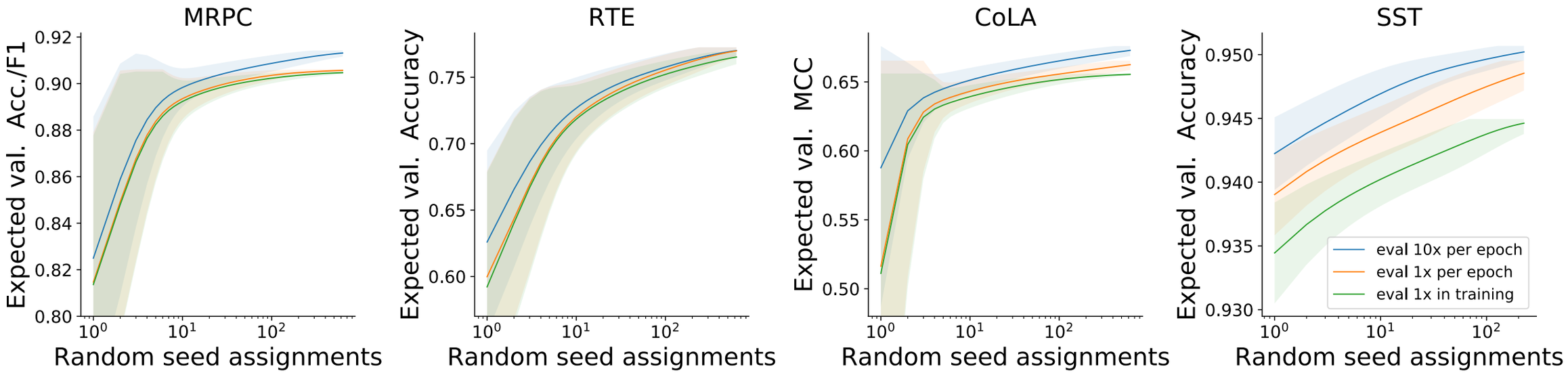


Figure 1. Expected validation performance (Dodge et al., 2019), plus and minus one standard deviation, as the number of experiments increases. The x-axis represents the budget (e.g., $x = 10$ indicates a budget large enough to train 10 models). The y-axis is the expected