Corpus Complexity Matters in Pretraining Language Models

Ameeta Agrawal, Suresh Singh

PortNLP lab, Department of Computer Science,

Portland State University

PortNLP

Portland State UNIVERSITY

Motivation

- Many studies show that more pretraining data leads to better performance in downstream NLP tasks, though this increases computational costs
- Some other studies show that increasing pretraining data does not always bring gains
- Yet other lines of research explore selecting appropriate data, reordering data, preprocessing or filtering data
- We ask: **given a fixed corpus budget, whether increasing the complexity of a training corpus yields higher performance more efficiently**

Problem Statement

- Let C be an unlabeled pretraining corpus of $|C|$ tokens and vocabulary $\mathrm{V_C}$
- Let D be a labeled downstream dataset of $|D|$ tokens and vocabulary $\mathrm{V_D}$

Problem Statement

- Given a fixed corpus budget (e.g., $|C|$ number of tokens), the goals are to:

  i. Construct corpora of distinct complexity

  ii. Measure similarity between these corpora and downstream datasets

  iii. Estimate correlation between complexity, similarity, and performance

Problem Statement

- Given a fixed corpus budget (e.g., $|C|$ number of tokens), the goals are to:

    i. Construct corpora of distinct complexity

    ii. Measure similarity between these corpora and downstream datasets

    iii. Estimate correlation between complexity, similarity, and performance

How to create corpora of different complexity?

- First we need a metric of estimating complexity at document d_i (or paragraph level)

- We use Flesch reading ease (FRE):

$$FRE(d_i) = 206.835 - 1.015 \left(\frac{\#words}{\#sents}\right) - 84.6 \left(\frac{\#syllables}{\#words}\right)$$

- Word and sentence lengths serve as proxies for semantic and syntactic complexity

- ↑ FRE scores == ↓ complexity (children's books)

- ↓ FRE scores == ↑ complexity (NYT article)

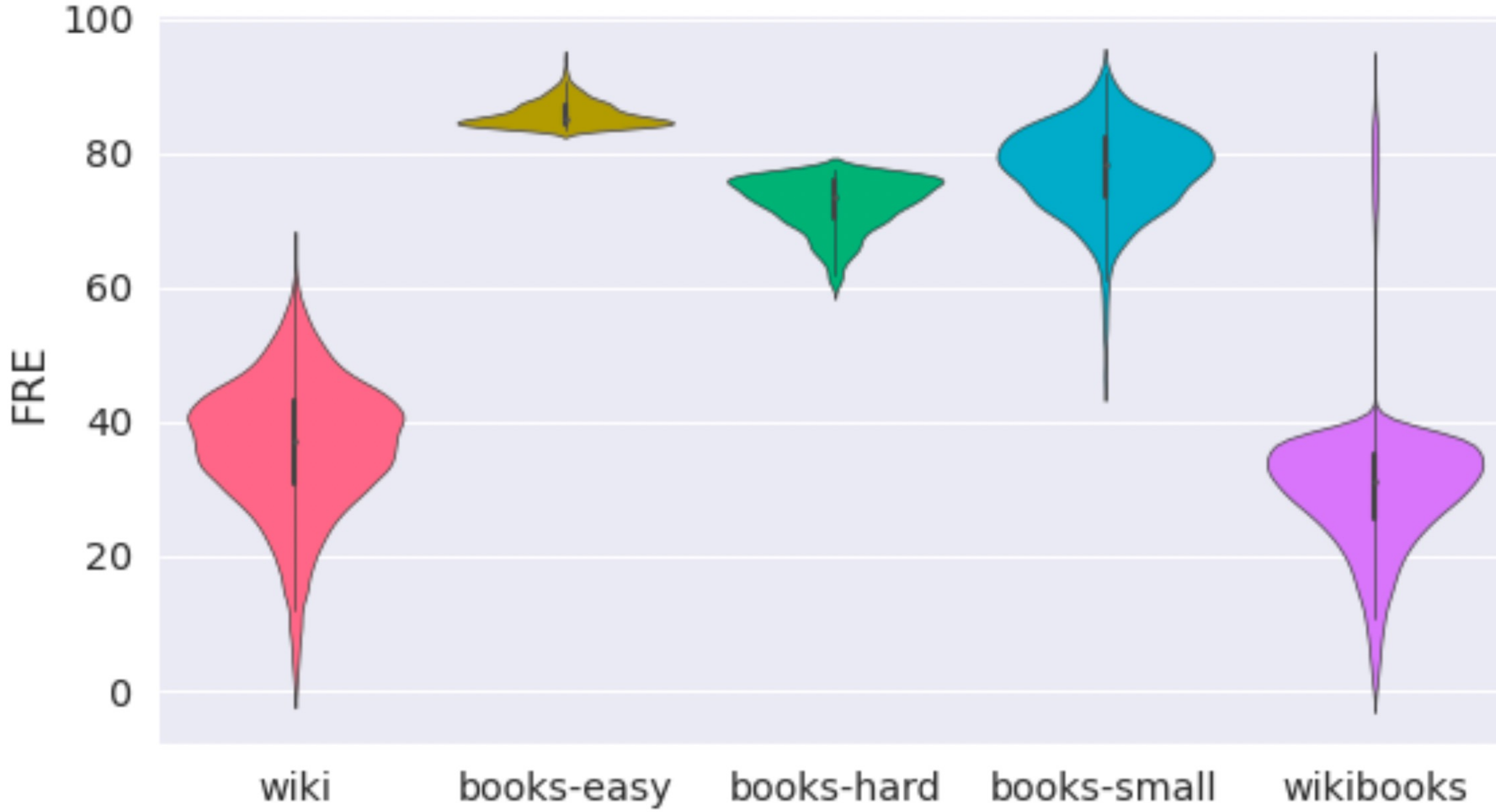
How to create corpora of different complexity?

- Next extract documents of different complexity from existing collections of text
- We choose two popular pretraining corpora:
  - Wiki-103
  - BookCorpus

How to create corpora of different complexity?

- Finally, we construct five corpora of different complexity, all of same size of ~100 million tokens:
  - **wiki**: the original Wiki-103 corpus (baseline)
  - **books-small**: random sampling of books from BookCorpus
  - **books-easy**: books of lowest complexity from BookCorpus
  - **books-hard**: books of hardest complexity from BookCorpus
  - **wikibooks**: blend of text of different levels of complexity

How to create corpora of different complexity?



FRE distribution of the corpora. *Lower* FRE indicates *higher* complexity. All corpora except **wikibooks** span narrow range of complexity.

Well, how do we confirm their complexity?

- There are established metrics for estimating lexical complexity at corpus level:
  - **Types**: number of unique tokens in a corpus (its vocabulary)
  - **Type-Token Ratio (TTR)**: function of vocabulary size and corpus size
  - **Entropy**: the greater the number of different words in a text, the higher its entropy

Well, how do we confirm their complexity?

Corpus	Tokens	Types	TTR (%)	Entropy
`wiki`	104M	267K	0.26	**7.375**
`books-easy`	120M	258K	0.22	6.294
`books-hard`	111M	417K	0.38	6.826
`books-small`	116M	346K	0.29	6.483
`wikibooks`	109M	**436K**	**0.40**	7.179

Characteristics of different pretraining corpora

- FRE can help create corpora of varying complexity.
- No corpus in our sample with entropy < 6 bits/word.

Problem Statement

- Given a fixed corpus budget (e.g., $|C|$ number of tokens), the goals are to:

    i. Construct corpora of distinct complexity

    ii. Measure similarity between these corpora and downstream datasets

    iii. Estimate correlation between complexity, similarity, and performance

How to measure similarity between corpus and downstream dataset?

- Two metrics:
  - Vocabulary Overlap Ratio (VOR): percentage of word types that appear in both sets of texts
  - Jensen-Shannon divergence (JSD): distance between two texts

Problem Statement

- Given a fixed corpus budget (e.g., $|C|$ number of tokens), the goals are to:

  i. Construct corpora of distinct complexity

  ii. Measure similarity between these corpora and downstream datasets

  iii. Estimate correlation between complexity, similarity, and performance

Implementation details

- Eight datasets from GLUE benchmark (CoLA, MNLI, MRPC, QNLI, QQP, RTE, SST-2, STS-B)
- Train from scratch several versions of BERT-base model
- Checkpoints saved after 10k, 20k, 30k steps
- Fine-tuned over downstream datasets for 2 epochs

Results and discussion

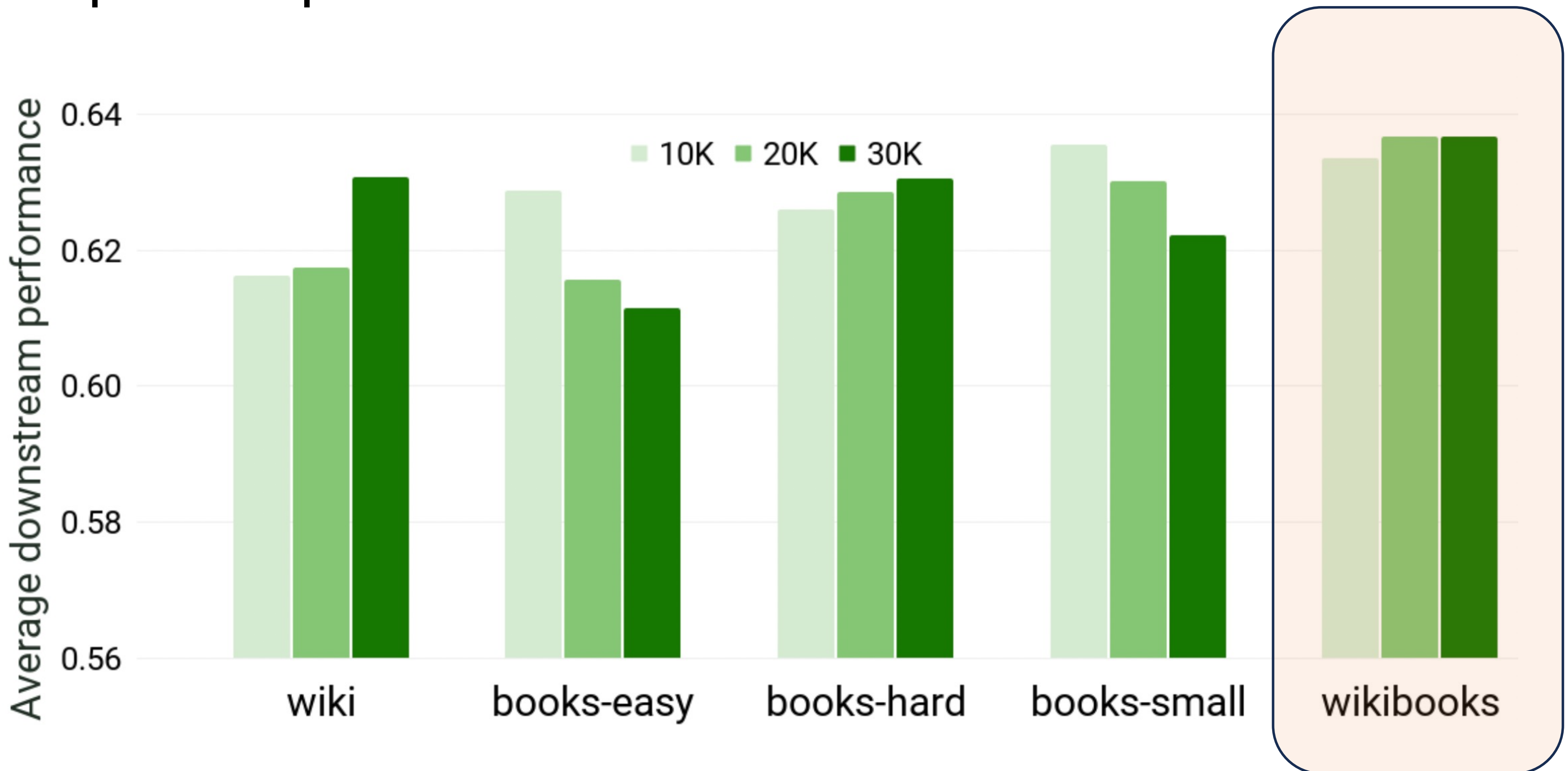
- We investigate:

    1. Whether a corpus of higher complexity leads to improved performance

    2. Whether a complex corpus is more similar to downstream data

    3. The correlation between complexity, similarity, and performance

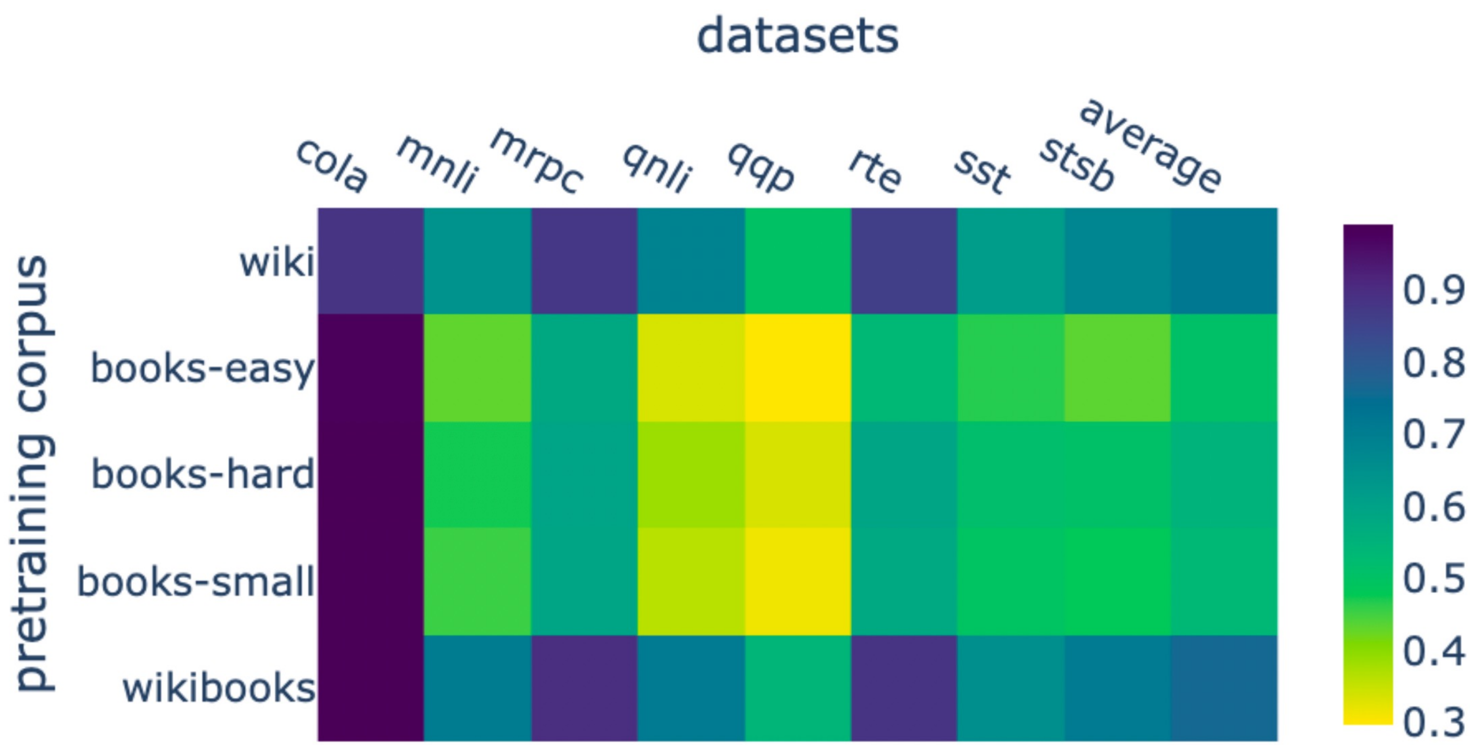
Whether a corpus of higher complexity leads to improved performance



Average GLUE score

- wikibooks performs best consistently
- Increased training does not always bring better performance (books-easy, books-small)

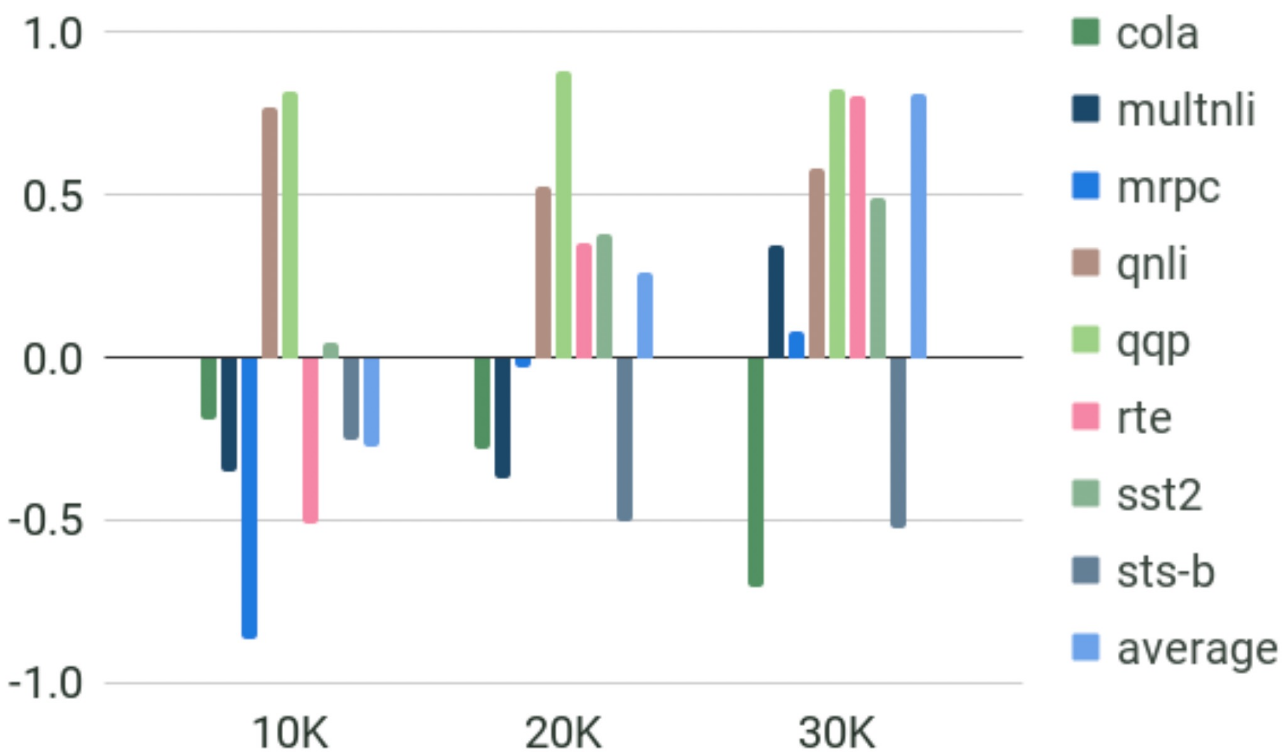
Whether a complex corpus is more similar to downstream data



Similarity (VOR) between pretraining corpus and downstream dataset

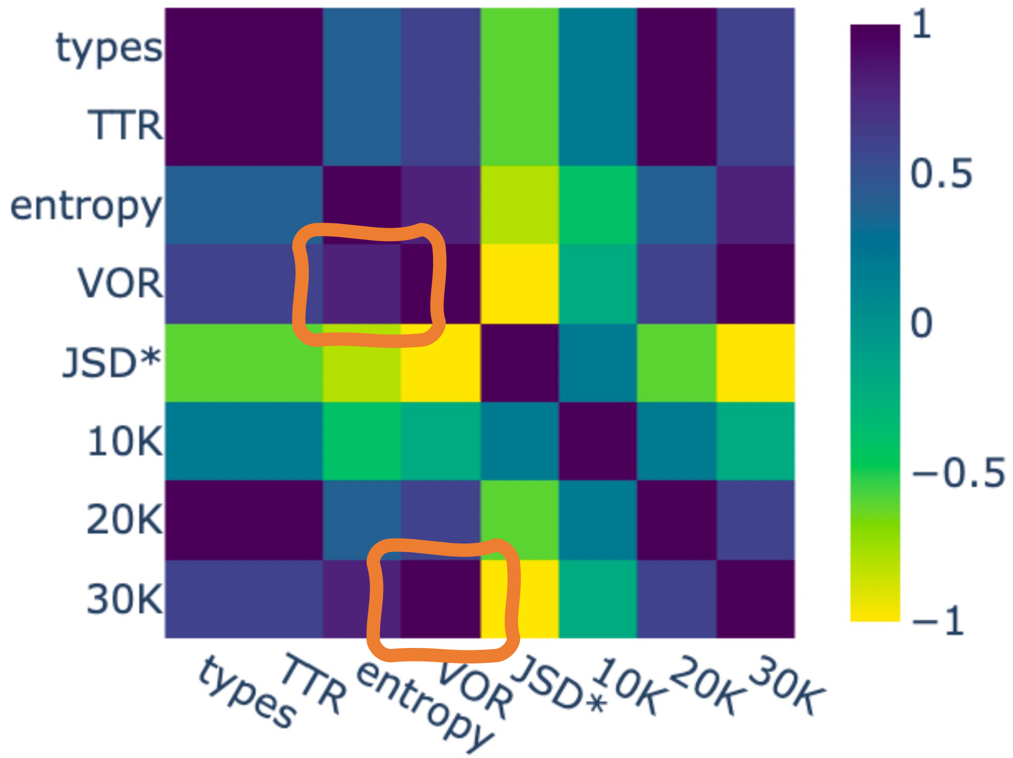
- wikibooks most similar to downstream datasets

Whether a complex corpus is more similar to downstream data



- Correlation between similarity and performance improves as training progresses

Correlation between complexity, similarity, and performance



- Performance (last row 30K) strongly correlated with VOR, which in turn correlates well with entropy

Conclusion

- FRE can help create corpora of varying complexity
- High complexity corpus (wikibooks) leads to highest performance
- wikibooks is also more similar to (GLUE) downstream datasets
- High correlation between similarity of corpus to downstream dataset, and corresponding performance, as well as with entropy
- Future work: explore the findings of this study in the context of generative (large) language models

Thank you!

Questions?

ameeta@pdx.edu

PortNLP

Portland State UNIVERSITY