Corpus Complexity Matters in Pretraining Language Models

Ameeta Agrawal and **Suresh Singh**
Department of Computer Science
Portland State University
{ameeta,singhsp}@pdx.edu

Abstract

It is well known that filtering low-quality data before pretraining language models or selecting suitable data from domains similar to downstream task datasets generally leads to improved downstream performance. However, the extent to which the quality of a corpus, in particular its complexity, affects its downstream performance remains less explored. In this work, we address the problem of creating a suitable pretraining corpus given a fixed corpus budget. Using metrics of text complexity we propose a simple yet effective approach for constructing a corpus with rich lexical variation. Our extensive set of empirical analyses reveal that such a diverse and complex corpus yields significant improvements over baselines consisting of less diverse and less complex corpora when evaluated in the context of general language understanding tasks.

1 Introduction

The recent trend in training language models (LM) has been to use increasingly larger text corpora (Khandelwal et al., 2019; Kaplan et al., 2020; Borgeaud et al., 2021). While this approach generally does improve downstream performance, it comes at a substantial computational cost. Another line of research has found that increasing the pretraining data does not always improve the performance on downstream tasks (Martin et al., 2019; Dai et al., 2019; Shin et al., 2022). In response, numerous studies have explored approaches such as utilizing pretraining corpora that are domain specific or using data filtering to reduce the size of the pretraining corpus, while improving downstream task performance (Beltagy et al., 2019; Lee et al., 2020; Grave et al., 2018; Raffel et al., 2019; Brown et al., 2020). The shortcoming of these methods is that the pretrained LM may be very specific to the selected tasks, and therefore, show limited generalizability to other downstream tasks, or require heuristic filtering techniques. In this research, we explore a complementary approach and investigate whether improving the complexity of the pretraining corpus can yield improved model performance. The implication is that rather than arbitrarily increasing the size of a corpus as is done today, increasing its complexity might yield higher performance but at a lower computational cost.

Intuitively it is easy to compare a children's book with a college textbook and state that the latter is more complex. Unfortunately, providing a general formal definition is fraught because books of different genres are complex in different ways (e.g., post-modern novel vs. biography). However, attempts have been made to characterize text complexity using reasonable measures such as vocabulary size, syntactic complexity, and semantic richness (Jensen, 2009). In this paper we use metrics that derive from these linguistic measures including types, type-token ratio, entropy, and Flesch reading-ease to estimate corpus complexity.

First we construct *five distinct corpora of equal size but varying complexity* to pretrain LMs. The resulting models are then fine-tuned and evaluated on downstream tasks from the GLUE benchmark. Our results suggest that a corpus containing a breadth of complexity from easy to hard but one that is skewed towards hard makes an effective corpus as evaluated in general language understanding tasks.

The key contributions of our paper are as follows: (i) We propose a simple approach for constructing a lexically rich and complex corpus for pretraining of language models; (ii) We conduct an extensive set of experiments by pretraining several language models from scratch on corpora of differing complexity, and then evaluating these models on a diverse set of downstream tasks; (iii) We analyze our results to estimate the correlation between the complexity of a corpus, its similarity to downstream data, and its performance on various downstream tasks.

Proceedings of The Fourth Workshop on Simple and Efficient Natural Language Processing (SustaiNLP), pages 257–263

2 Related Work

Below, we briefly review two broadly related threads of research.

Data selection. Ruder and Plank (2017) proposed several similarity and diversity measures for assessing the suitability of data for transfer learning. Dai et al. (2019) studied the problem of selecting appropriate corpus for pretraining in the context of Named Entity Recognition (NER) downstream tasks, and found that language models pretrained on source text similar to the target task outperform the ones pretrained on other sources (with one exception). Gururangan et al. (2020) compared the vocabulary overlap between pretraining sources and target domain corpora, and found that the pretrained model performs slightly better when target domain is less distant than source domain, but not in all the cases. Lange et al. (2021) studied the selection of source data for transfer learning.

Selecting data from similar domains as downstream tasks for pretraining of domain-specific language models has generally been shown to be beneficial, e.g., SciBERT (Beltagy et al., 2019), BioBERT (Lee et al., 2020). However, prior work has also observed that this trend does not always hold true (Martin et al., 2019; Shin et al., 2022). Dai et al. (2020) found that models pretrained on forums corpus (0.6B tokens) outperformed those trained on tweets corpus (0.9B tokens) on both forums- and tweets-related downstream tasks, as well as a significantly larger generic BERT model (3.3B tokens), highlighting the importance of domain similarity of corpus over its size.

Data engineering. A complementary line of research suggests that engineering the corpus before pretraining through reordering (Agrawal et al., 2021; Nagatsuka et al., 2021; Li et al., 2021; Wang et al., 2023), preprocessing (Babanejad et al., 2023), and filtering (Grave et al., 2018; Raffel et al., 2019; Brown et al., 2020; Rae et al., 2021; Kreutzer et al., 2022) can potentially enhance both the overall performance and efficiency of language models.

Diverging from previous studies, our research focuses on examining the influence of the *complexity* of a pretraining corpus on downstream tasks related to general language understanding. To accomplish this, we introduce a straightforward methodology for constructing a corpus that embodies richness and complexity.

3 Method

Let $\mathcal{C}$ be an unlabeled pretraining corpus of $|\mathcal{C}|$ total tokens, consisting of a vocabulary set $V_{\mathcal{C}}$, i.e., the unique tokens or types in $\mathcal{C}$. Similarly, let $\mathcal{D}$ be a labeled downstream dataset with total number of tokens $|\mathcal{D}|$ and a vocabulary set $V_{\mathcal{D}}$. Given a fixed corpus budget (e.g., number of tokens), we first aim to construct distinct corpora of various complexity. Then, the goal is to measure the similarity between these corpora and downstream datasets, and estimate the correlation between complexity, similarity, and performance.

We present some metrics for assessing the complexity of a corpus and for computing the similarity between two collections of text – the pretraining corpus and the downstream datasets in subsections 3.1 and 3.2, before describing the procedure for creating corpora of varying complexity in subsection 3.3.

3.1 Corpus Complexity

We consider three metrics for estimating the complexity of a text corpus.

Types. This is the number of types or unique tokens in a corpus (i.e., its vocabulary).

Type-Token Ratio (TTR). Lexical complexity can also be indexed via TTR – the higher the ratio, the greater the lexical diversity in the sample (Johnson, 1944). Although TTR is often sensitive to length of the texts, for analyzing corpora of comparable sizes, it can serve as a useful metric (Johansson, 2008), and is computed as $TTR(\mathcal{C}) = \frac{|V_{\mathcal{C}}|}{|\mathcal{C}|}$.

Entropy. Broadly speaking, entropy is a measure of randomness or disorder (Shannon, 1948; Fano, 1961), and the greater the number of different words in a text, the higher its entropy, or, conceptually, its complexity. We calculate the unigram entropy of $\mathcal{C}$ as follows:

$$H(\mathcal{C}) = -\sum_{i=1}^{|V_{\mathcal{C}}|} p(w_i) \log_2 p(w_i)$$

where $p(w_i)$ is the probability of type w_i in $\mathcal{C}$.

3.2 Text Similarity

We adopt two well-defined measures to estimate the similarity between two pieces of text, such as the pretraining corpus $\mathcal{C}$ and a downstream dataset $\mathcal{D}$.

Vocabulary Overlap Ratio (VOR). This computes the percentage of word types that appear in both the texts ($V_{\mathcal{C}}$ and $V_{\mathcal{D}}$) where a higher ratio indicates higher similarity, and is calculated as:

$$VOR(\mathcal{C}, \mathcal{D}) = \frac{|V_{\mathcal{C}} \cap V_{\mathcal{D}}|}{|V_{\mathcal{D}}|}.$$

Jensen-Shannon divergence (JSD). This metric measures the distance between two texts (Lin, 1991), and $D^{(JS)}$ is defined as:

$$D^{(JS)}(P||Q) = \alpha_1 D^{(KL)}(P||M) + \alpha_2 D^{(KL)}(Q||M)$$

where $M = \alpha_1 P + \alpha_2 Q$, and P and Q are the probability distributions of two texts (e.g., a pretraining corpus $\mathcal{C}$ and a downstream dataset $\mathcal{D}$, in our case). The values of α_1 and α_2 are set as 0.5 each. $D^{(KL)}$ is Kullback-Leibler divergence, a measure for comparing the differences in two texts, and is defined as, $D^{(KL)}(P||Q) = \sum_i p_i \log \frac{p_i}{q_i}$.

3.3 Constructing Corpora with Varying Complexity

The complexity of a corpus can be summarized by using metrics like number of types, type-token ratio, and entropy (section 3.1). However, in order to create a corpus according to varying complexity we need a more fine-grained metric that can *compute complexity at document (or even paragraph) level*. One such metric is the Flesch reading ease (FRE) score, commonly used to assess the difficulty of a piece of text (Flesch, 1948).

For a document $d_i \in \mathcal{C}$, its FRE score is computed as:

$$FRE(d_i) = 206.835 - 1.015\left(\frac{\#w}{\#s}\right) - 84.6\left(\frac{\#l}{\#w}\right)$$

where $\#w$, $\#s$, and $\#l$ denote the number of words, sentences, and syllables in d_i, respectively. The word and sentence length serve as proxies for semantic and syntactic complexity, respectively. Note that texts with high FRE scores tend to display lower complexity (e.g., children's books), while an editorial in the New York Times which has a much greater complexity, shows lower FRE scores. Thus, our approach for creating a more complex corpus is to combine pieces (paragraphs or documents) from existing corpora based on their FRE score.

Our method starts by adopting two text corpora widely used for pretraining of language models:

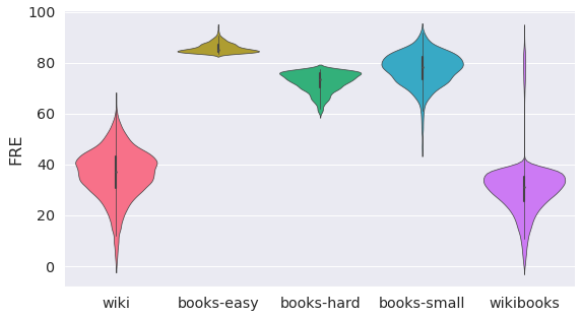


Figure 1: FRE distribution of the corpora. *Lower* FRE indicates *higher* complexity. `wikibooks` spans the full spectrum of complexity, consisting of both low and high complexity, but mostly skewed towards the latter.

wiki-103, a subset of English Wikipedia (Merity et al., 2016) and **BookCorpus**, a large collection of books (Zhu et al., 2015). From these, we construct the following five corpora:

- `wiki`: This is the original wiki-103 corpus consisting of around 100 million tokens.
- `books-small`, `books-easy`, `books-hard`: Next, we create a comparably-sized corpus of ∼100M tokens, called `books-small`, by randomly sampling books from BookCorpus. Then, for each book in BookCorpus, we compute its FRE score and create two relevant baselines: `books-easy` by combining books of lowest complexity (i.e., the highest FRE scores), and conversely, `books-hard` by using books with the highest complexity (i.e., the lowest FRE scores).
- `wikibooks`: Finally, we hypothesize that a complex and diverse corpus contains a *blend* of texts with different levels of complexity, albeit with a focus on more complex ones. We speculate that this composition would allow it to capture the nuanced linguistic aspects present in a wide range of texts. To create such a corpus, which we call `wikibooks`, we first sample some articles from wiki-103 and books from BookCorpus of varying complexity (i.e., FRE scores ranging from high to low), and then use up the remaining corpus quota by sampling texts of mostly high complexity (low FRE scores).

Figure 1 plots the FRE distribution of each of the five corpora. As we can see, `books-easy`,

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

Table 1: Characteristics of different pretraining corpora.

books-hard, and books-small span a narrow range of complexity all skewing towards less complex; wiki has moderate to high complexity; and wikibooks is the only one to show the broadest range of complexity, with most of the mass concentrated in the highest complexity range, but also some in the lowest complexity range.

3.4 Downstream Datasets and Implementation

We use eight datasets from the General Language Understanding Evaluation (GLUE) benchmark in our experiments, which includes CoLA, MNLI, MRPC, QNLI, QQP, RTE, SST-2 and STS-B (Wang et al., 2018).

Text tokenization is done using NLTK[1], and FRE scores are computed using Readability package[2]. Using the different corpora, we pretrain from scratch different versions of BERT-base model[3] (Devlin et al., 2019). The training continues for at most 30K steps. Checkpoints saved after 10K, 20K, and 30K steps are then fine-tuned over the downstream datasets for two epochs each.

4 Discussion

Our work investigates: (i) whether document-level metric such as FRE can be used to construct corpora of varying complexity, (ii) whether corpora of higher complexity lead to improvements in downstream performance, (iii) whether a complex corpus is more similar to downstream data, and (iv) the correlation between complexity, similarity, and performance.

[1]We use NLTK tokenizer: https://www.nltk.org/api/nltk.tokenize.html.

[2]We use Readability package: https://pypi.org/project/readability/ To account for the length-based differences in Wikipedia articles and Books, we randomly but sequentially select a subset of 1000 sentences for each book when computing its FRE.

[3]We use the uncased version, with 12 transformer layers, batch size set to 8, maximum length of the input sequence set to 512, and all other settings set as default. All pretraining and fine-tuning experiments are performed using HuggingFace library (Wolf et al., 2019).

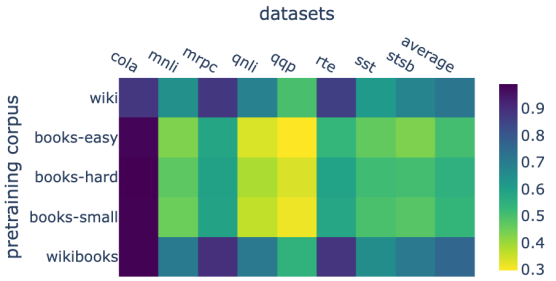
Figure 2: Comparison of (unweighted) average GLUE score, across five different pretraining corpora under varying number of training steps (10K, 20K, 30K).

Whether FRE can help create suitably complex corpus. Table 1 summarizes the details of the five distinct corpora, where we find that wikibooks, which contains a mix of low and high complexity text, has the highest number of types and TTR, and second highest entropy. This demonstrates the effectiveness of using a computationally simple metric such as FRE in creating corpora of a wide range of complexity. Moreover, we also notice that there is no corpus in our sample with a unigram entropy of less than six bits/word, which is in line with information-theoretic models of communication (Bentz et al., 2017).

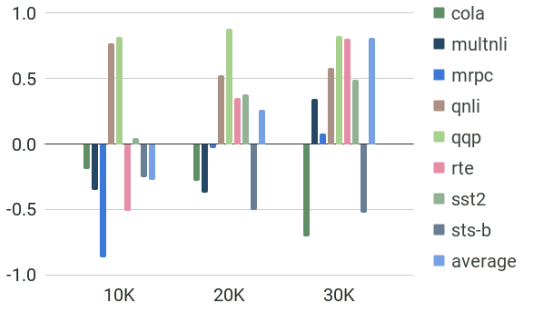
Analyzing corpus complexity and downstream performance. Figure 2 plots the average scores across eight downstream tasks obtained using models pretrained with the five different corpora under varying number of training steps. Three out of five corpora yield increasingly better results as the training progresses, except books-easy and books-small which show the opposite trend. On the one hand, this suggests that simply training for longer time does not always guarantee a monotonically increasing performance score. On the other hand, this also indicates that pretraining on fairly less complex corpora (cf. Fig. 1) is generally less effective.

In connecting the results of Figure 2 with complexity metrics reported in Table 1, we observe that wikibooks, a corpus with a comparatively higher degree of complexity characterized by a larger number of word types and a higher TTR, consistently outperforms all other corpora across the three model checkpoints. On the opposite end is the poorest performing corpus books-easy with the fewest types, lowest TTR, and lowest entropy.

Analyzing similarity between pretraining corpus and downstream datasets. Now, we assess the similarity between these corpora and downstream



(a) Similarity (VOR) between pretraining corpus and downstream dataset (darker shades indicate higher similarity)



(b) Correlation between similarity (VOR) and performance (positive correlation is better)

Figure 3: (**top**) Similarity (VOR) between pretraining corpora and downstream datasets (train). (**bottom**) Pearson's correlation analysis (similarity and performance).

datasets to examine whether a more complex corpus provides greater alignment with the downstream data. Figure 3a shows that `wikibooks` is more similar to all the downstream datasets in comparison to the other corpora, aligning with the intuition that a corpus with richer vocabulary subsequently has increased similarity with downstream data. As a further analysis, Figure 3b shows a moderate to high correlation between the similarity of the corpus to downstream datasets and the corresponding performance across most datasets, which strengthens as training progress. Similar trends hold for JSD (included in Appendix A). These findings indicate that pretraining using a corpus that is similar to the downstream datasets is generally beneficial, and VOR provides a computationally simple way of estimating this similarity.

Analyzing complexity, similarity, and performance. Figure 4 presents Kendall's Tau correlation analysis for all three factors: complexity, similarity, and performance. In looking at the last row in particular (i.e., performance of the '30K' model) we observe that performance is strongly correlated with VOR, which in turn is strongly correlated with

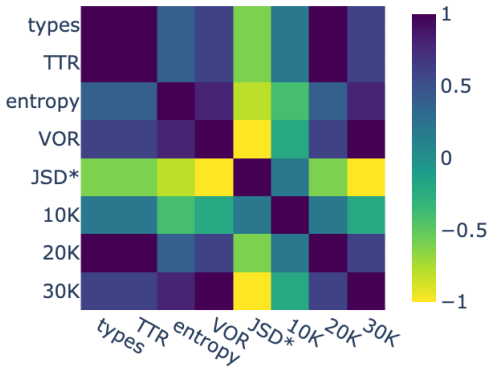


Figure 4: Kendall's Tau analysis comparing performance, complexity, and similarity. Darker shades indicate better correlation except for JSD, where a lighter shade (negative correlation) is desirable.

metrics of complexity (types, TTR, and entropy). Taken together, these results suggest that a more complex corpus leads to better downstream evaluation performance.

5 Conclusions

We investigate whether pretraining on a corpus with higher complexity subsequently yields improved performance in downstream evaluations. Within this study, we construct corpora of diverse complexities by using straightforward metrics like Flesch reading ease, and estimate corpus-level complexity using metrics such as unique word types, type-token ratio, or unigram entropy. The results of our extensive empirical analysis, which involves training language models from scratch using five distinct corpora of varying text complexity and evaluating their performance across eight downstream tasks, suggest a strong correlation between corpus complexity, its similarity to downstream data, and the resulting performance on these tasks. One interesting direction for future research involves exploring the findings of this study in the context of generative language models.

Limitations

One limitation of our study is that, due to computational constraints, we use what are now considered as relatively "small-sized" models and corpora, exclusively focusing on the English language and generic domains such as Wikipedia articles and books. The generalizability of our findings to larger corpora, other languages, or specific domains such as medical texts warrants further investigation.

Acknowledgments

We thank the anonymous reviewers for their insightful comments. This work was partially supported by NSF grants 2246174 and 1910655.

References

Ameeta Agrawal, Suresh Singh, Lauren Schneider, and Michael Samuels. 2021. On the role of corpus ordering in language modeling. In *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pages 142–154.

Nastaran Babanejad, Heidar Davoudi, Ameeta Agrawal, Aijun An, and Manos Papagelis. 2023. The role of preprocessing for word representation learning in affective tasks. *IEEE Transactions on Affective Computing*, pages 1–18.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*.

Christian Bentz, Dimitrios Alikaniotis, Michael Cysouw, and Ramon Ferrer-i Cancho. 2017. The entropy of words—learnability and expressivity across more than 1000 languages. *Entropy*, 19(6):275.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2021. Improving language models by retrieving from trillions of tokens. *arXiv preprint arXiv:2112.04426*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Xiang Dai, Sarvnaz Karimi, Ben Hachey, and Cecile Paris. 2019. Using similarity measures to select pretraining data for NER. *arXiv preprint arXiv:1904.00585*.

Xiang Dai, Sarvnaz Karimi, Ben Hachey, and Cecile Paris. 2020. Cost-effective selection of pretraining data: A case study of pretraining bert on social media. *arXiv preprint arXiv:2010.01150*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Robert M Fano. 1961. Transmission of information: A statistical theory of communications. *American Journal of Physics*, 29(11):793–794.

Rudolph Flesch. 1948. A new readability yardstick. *Journal of applied psychology*, 32(3):221.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. *arXiv preprint arXiv:1802.06893*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Kristian TH Jensen. 2009. Indicators of text complexity. *Mees, IM; F. Alves & S. Göpferich (eds.)*, pages 61–80.

Victoria Johansson. 2008. Lexical diversity and lexical density in speech and writing: A developmental perspective. *Working papers/Lund University, Department of Linguistics and Phonetics*, 53:61–79.

Wendell Johnson. 1944. Studies in language behavior: A program of research. *Psychological Monographs*, 56(2):1–15.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2019. Generalization through memorization: Nearest neighbor language models. *arXiv preprint arXiv:1911.00172*.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, et al. 2022. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.

Lukas Lange, Jannik Strötgen, Heike Adel, and Dietrich Klakow. 2021. To share or not to share: Predicting sets of sources for model transfer learning. *arXiv preprint arXiv:2104.08078*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Conglong Li, Minjia Zhang, and Yuxiong He. 2021. Curriculum learning: A regularization method for efficient and stable billion-scale gpt model pre-training. *arXiv preprint arXiv:2108.06084*.

Jianhua Lin. 1991. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de La Clergerie, Djamé Seddah, and Benoît Sagot. 2019. Camembert: a tasty french language model. *arXiv preprint arXiv:1911.03894*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*.

Koichi Nagatsuka, Clifford Broni-Bediako, and Masayasu Atsumi. 2021. Pre-training a bert with curriculum learning by increasing block-size of input text. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 989–996.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Sebastian Ruder and Barbara Plank. 2017. Learning to select data for transfer learning with Bayesian optimization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 372–382, Copenhagen, Denmark. Association for Computational Linguistics.

Claude Elwood Shannon. 1948. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423.

Seongjin Shin, Sang-Woo Lee, Hwijeen Ahn, Sungdong Kim, HyoungSeok Kim, Boseop Kim, Kyunghyun Cho, Gichang Lee, Woomyoung Park, Jung-Woo Ha, et al. 2022. On the effect of pretraining corpora on in-context learning by a large-scale language model. *arXiv preprint arXiv:2204.13509*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

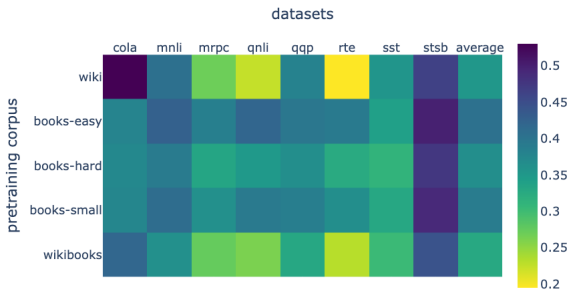
Yile Wang, Yue Zhang, Peng Li, and Yang Liu. 2023. Language model pre-training with linguistically motivated curriculum learning.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

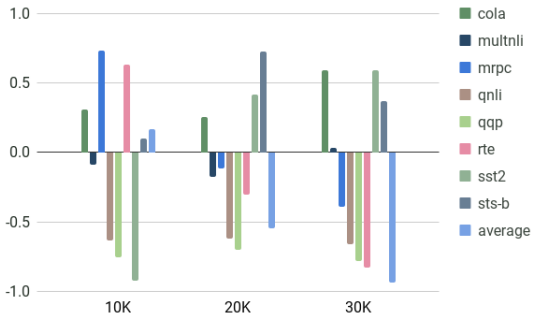
Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

A Similarity Analysis

Figure 5 presents the results of similarity analysis and Pearson's correlation analysis using Jensen-Shannon divergence.



(a) JSD (lighter shades indicate higher similarity)



(b) JSD (negative correlation is better)

Figure 5: (**top**) Similarity between pretraining corpora and downstream datasets (train set) using JSD. The last column 'average' presents the average results of all the datasets. (**bottom**) Pearson's correlation analysis between JSD and performance at 10K, 20K, and 30K step checkpoints.