Hints on the data for language modeling of synthetic languages with transformers

Rodolfo Zevallos[1] **and Núria Bel**[1]
Universitat Pompeu Fabra
Barcelona, Spain
rodolfojoel.zevallos@upf.edu, nuria.bel@upf.edu

Abstract

Language Models (LM) are becoming more and more useful for providing representations upon which to train Natural Language Processing applications. However, there is now clear evidence that attention-based transformers require a critical amount of language data to produce good enough LMs. The question we have addressed in this paper is to what extent the critical amount of data varies for languages of different morphological typology, in particular those that have a rich inflectional morphology, and whether the tokenization method to preprocess the data can make a difference. These details can be important for low-resource languages that need to plan the production of datasets. We evaluated intrinsically and extrinsically the differences of five different languages with different pretraining dataset sizes and three different tokenization methods for each. The results confirm that the size of the vocabulary due to morphological characteristics is directly correlated with both the LM perplexity and the performance of two typical downstream tasks such as NER identification and POS Tagging. The experiments also provide new evidence that a canonical tokenizer can reduce perplexity by more than a half for a polysynthetic language like Quechua as well as raising macro-F1 score from 0.8 to more than 0.9 in both downstream tasks with a LM trained with only 6M tokens.[1].

1 Introduction

Language Models (LMs) are becoming more and more useful for providing representations upon which to train different Natural Language Processing (NLP) applications. However, there is evidence that LMs trained with attention-based transformers need large quantities of pretraining language data to provide good enough representations that can be used in downstream tasks.

To have very large amounts of data, multilingual LMs have been proposed as a solution. However, there is evidence (Rust et al., 2021, Bansal et al., 2021, Goyal et al., 2021) that the monolingual LMs outperformed their multilingual counterparts.

As for the amount of monolingual data required, Zhang et al. (2021) experiments with English showed that the amount of data for reaching at least an 80% average over several tasks of relative performance is around 10M tokens. The question we have addressed in our research is whether the critical figures for English are the same for other languages and in particular for languages of a different morphological type. Having hints about the critical amount of data and tokenization strategies to make the most profit of the available data is of upmost importance for low-resource languages, many of them with a morphology more complex than that of English, and that need to plan the production of datasets.

A LM is an assessment of the probability distribution over sequences of words given a fixed set of words with parameters estimated from data. The increase in the number of tokens of the vocabulary of particular languages due to their inflectional morphology has been demonstrated to affect the coverage of the Markovian LMs (Whittaker and Woodland, 2003). For current attention-based transformer language models (TLM), like RoBERTa that is a closed vocabulary system, the direct consequence of modeling a rich inflectional morphology should also be that the coverage of the vocabulary will be lower than that of a morphologically simpler language. For instance, Mielke et al. (2019) found that English was among the easiest languages for building a LM, while German, which is a synthetic language, was among the hardest. Polysynthetic languages like Quechua, with more than 100 inflectional suffixes, and in which up to five suffixes can be attached to a verbal stem, would have harder modeling problems that will aggravate its problems

[1]Equal contribution

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 12508–12522

for being a low-resource language.

To understand how the amount of critical pretraining data varies for different languages, we reproduced Zhang et al. (2021) experiments but for different languages of an increasing degree of morphological complexity, as measured by type-token ratio (TTR) following Kettunen (2014) and Mielke et al. (2019). The languages are: English, French, German, Turkish and Quechua. In Table 1, the TTR of these languages assessed with the 6M datasets used in our experiments shows the big differences among these languages.

Language	Type	Tokens	TTR
English	132,936	6,000,198	0.0221
French	188,741	6,000,003	0.0314
German	201,465	6,000,086	0.0335
Turkish	262,531	6,000,093	0.0437
Quechua	325,248	5,985,472	0.0543

Table 1: Number of Tokens, Type-Tokens and Type-Token Ratio (TTR) for each language for the 6M dataset

However, we reproduced the conditions of Zhang et al. (2021) but with datasets of 1M, 2M, 3M, and 6M for each language, as Quechua has no more corpus available. For all languages and datasizes we carried out an intrinsic evaluation, i.e. differences in LM perplexity and an extrinsic evaluation, i.e. to assess to what extent critical learning can be achieved with representations made with smaller datasets. We have used the representations produced by the different models to fine-tune classifiers for Name Entity Recognition (NER) and Part-of-Speech (POS) tagging.

Besides, we repeated the different size experiments with three different tokenization methods, to get evidence on whether a linguistically motivated tokenizer improves both perplexity and classification results. We have compared three segmenters that produce subword tokenization: BPE (Sennrich et al., 2016), Unigram (Kudo, 2018) and DeepSpin (Peters and Martins, 2022). BPE is one of the most used tokenizers nowadays. It initially segments the text into characters and then it iteratively merges together the most frequently co-occurring symbols until finding space boundaries or reaching a previously set vocabulary limit. Unigram works by segmenting the texts into words following space boundaries to build an initial vocabulary, and trimming down each symbol to obtain a shorter vocabulary list. Differently to BPE and Unigram, DeepSpin is a supervised canonical tokenizer. Mager et al. (2020) introduced canonical segmentation as a morphological segmentation that consists of dividing words into their standardized morphemes. A canonical tokenizer attempts to recompose the character sequence that suffers some modification when concatenated or combined with other morphemes. For instance, in English 'profitable' becomes 'profitably' when combined with the adverbial morpheme 'ly'. Canonical tokenization should produce the following tokens: 'profitable' and 'ly', therefore reducing the vocabulary size considerably.

The contributions of our research are two. First, an evaluation of the critical amount of data for training a performant TLM. The evaluation is done intrinsically in terms of perplexity, and extrinsically by using the produced representations to fine-tune classifiers for two downstream applications: POS tagging and NER. Second, evidence, both from the intrinsic and the extrinsic evaluations, about how much a linguistically motivated tokenization maximizes the profit of small datasets. These hints might be crucial for keeping technologically alive languages that cannot get the exorbitant amount of textual data that ensures maximal performance. Besides, it is also important to get more understanding about the capabilities of methods that could significantly differ when used for languages other than English.

2 Related work

Hu et al. (2020) and Warstadt et al. (2020) were the first papers addressing the amount of data necessary for training large LM. Hu et al. (2020) trained four classes of neural models and one baseline n-gram model on four datasets derived from a newswire corpus, consisting of 1M, 5M, 14M, and 42M, to assess differences in syntactic probing tasks among different architectures and pretraining corpora sizes. The main outcome of their experiments was to find out that perplexity of the LM and performance in the addressed probing tasks did not correlate; that is, LM trained with more data, and therefore lower perplexity, were not better at the probing tasks. They concluded that the architecture proved to be a more important source of differences than the size of the dataset, with the GPT-2 transformer, using BPE, achieving best results.

Warstadt et al. (2020) pretrained 12 RoBERTa (Liu et al., 2019) models on English corpora

varying in size and tokenized with BPE. These MiniBERTa models were trained with quantities of data ranging of 1M, 10M, 100M, 1B. The results showed that RoBERTa learns linguistic features with only a few million words, but that it takes billions of words for the model to prefer to use linguistic generalizations over surface ones. Using the same models, Zhang et al. (2021) explored the relation between the amount of data and the effectiveness of RoBERTa for learning grammatical features and other linguistic phenomena of English. They performed an extensive collection of tests for showing the learning curve on different tasks of the miniBERTa models pretrained with data of different size, from 1M to 1B words. Their results show that the learning for traditional NLP tasks such as POS labeling, NER identification and other higher level tasks dealing with syntax and semantics occur with less than 100M words of pretraining data. In particular, learning for POS tagging and NER is reported to happen with about 10M words, having no big further improvements after that. Pérez-Mayos et al. (2021) also used the MiniBERTas models developed by Warstadt et al. (2020) to explore the relation between the size of the pretraining data and the syntactic capabilities of RoBERTa. For all the tasks studied, the models with more training data performed better, however the performance improvement growth was also stalled after 10M for tasks like POS tagging.

For languages other than English, Micheli et al. (2020) worked on French texts with CamemBERT (Martin et al., 2020) that is similar to RoBERTa but uses whole-word masking and SentencePiece tokenization (Kudo and Richardson, 2018), which uses Unigram, and different pretraining data sizes. Their results showed that 100 MB of raw text (about 10,5 M words) were sufficient to reach a similar performance than with larger datasets on a question answering task. Micallef et al. (2022) found that 46M tokens of pretraining were enough for a Maltese BERT to be comparable with a multilingual BERT adapted with vocabulary augmentation methods. Inoue et al. (2021) worked on assessing the impact of language variants, data sizes and fine-tuning tasks with Arabic pretrained TLM. They trained 8 Arabic models, named CAMeLBERT of 6.3B, 3.1B, 1.5B, and 636M words, that were evaluated on different NLP tasks including NER and POS tagging. They concluded that the amount of pretraining data had limited and inconsistent effects on the performance of the fine-tuned classifiers. However, note that the size of the datasets in these experiments were far beyond the 10M that Warstadt et al. (2020) or Micheli et al. (2020) identified as the amount from which the model seems unable to learn more.

The relation of the morphological type and the robustness of language models because of the size of the vocabulary is a well known topic. A high number of words in the vocabulary is a characteristic of languages of a higher morphological complexity due to inflectional and derivational processes. For instance, Quechua, which is a polysynthetic language, typically has 3 morphemes per word and about 100 different suffixes, while English has around 1.5 morphemes per word, and about 35 suffixes. Geutner (1995) was one of the first works to afford evidence on reducing about 50% perplexity in a statistic language model by using a morpheme-based n-gram model for the task of speech recognition of German. German, in addition to inflection morphology, uses prefixes to create new verbal tokens: *ausgehen* ('to go out'), *hineingehen* ('to go in') and noun-noun composition is extremely frequent with an, in principle, unlimited number of nouns being concatenated creating new nouns. According to Geutner (1995), morpheme-based n-gram models proved to get more robust probability estimates with smaller training datasets and also limited the size of the vocabulary.

Mielke et al. (2019) studied whether there are typological properties that make certain languages harder to language model than others, and studied linguistic features that correlated to difficulties for creating a LM. They reported on language modeling results on 62 languages from 13 language families using Bible translations, and on the 21 languages used in the European Parliament proceedings. They conducted a correlational study of features of a language to find one that is predictive of modeling difficulty. Their results confirmed that the type inventory or vocabulary size is a statistically significant indicator of the modeling difficulty. Park et al. (2021) revisited these results and performed experiments for 92 languages also from a corpus of Bibles. Their results confirmed that number of types or size of the vocabulary of the related TTR, are statistically correlated to difficulties for language modeling. Additionally, the research was extended for assessing how different segmentation methods captured morphological segments

and the impact of tokenization in the final results. The results were that subword tokenization methods outperformed character-level ones. BPE was reported to fail mitigating the problems created by languages with high TTR, while other segmenters that were informed with linguistic information did better.

The gains achieved by linguistically motivated tokenization were also observed in other research areas like Machine Translation. Rust et al. (2021) empirically compared multilingual pretrained language models to their monolingual counterparts on a set of nine typologically diverse languages. They concluded that while the pretraining data size is an important factor, the tokenizer of each monolingual model plays an equally important role in the performance on downstream tasks. The results indicate that the models trained with dedicated monolingual tokenizers outperform their counterparts with multilingual tokenizers in most tasks. While the smallest performance gap is for POS tagging (at most 0.4% accuracy), performance gap for NER reaches even 1.7 difference in macro-F1 score for Arabic. Ortega et al. (2020), Chen and Fazio (2021), and Mager et al. (2022) are works comparing different tokenizers for improving translation in low-resource language pairs. Their results provided evidence that a linguistically motivated segmentation leads to significant improvements in translation quality specially in low-resource contexts.

3 Methodology

In our experiments, we tested 20 RoBERTa models. We pretrained from scratch LM for English, German, French, Turkish and Quechua with different pretraining datasets ranging from 1M to 6M tokens, and we used three different tokenizers for each: BPE, Unigram and DeepSpin. Code and resources are available on https://github.com/IULATERM-TRL-UPF/Hints-on-the-data-for-language-modeling

3.1 Pretraining

3.1.1 Pretraining Data

We pretrained RoBERTa models for the mentioned five languages, following the same conditions of Warstadt et al. (2020) trained the miniBERTas models for English, but further reducing the size of datasets. The training data used in our pretraining of RoBERTa are the following.

For English, a random part of the Wikipedia corpus of 2.5 billion tokens used by Devlin et al. (2019) to train BERT. For German, French and Turkish, we used parts of OSCAR corpora extracted from the Common Crawl November 2018 snapshot, automatically classified for language identification and filtered to avoid noise (Ortiz Suárez et al., 2019). The German OSCAR, with 21 billion tokens, was the one used by Scheible et al. (2020), the French OSCAR, with 32.7 billion tokens, the one that was used by Martin et al. (2020) and the Turkish OSCAR with 11.5 million documents, the one that was used by Toraman et al. (2022). Monolingual-quechua-iic[2] in Quechua (6 million tokens) used by Zevallos et al. (2022). This Quechua corpus is composed of a wide variety of sources, including Wikipedia (about 1 million tokens) and other resources available on the Internet, as well as educational materials and legal documents. For each language, we randomly produced training sets with a total of 1M, 2M, 3M, and 6M tokens each.

3.1.2 Tokenization

For our experiments we compared three different tokenizers: BPE, Unigram and DeepSpin. Similar to the experiments performed by Liu et al. (2019) to train RoBERTa, we used BPE (Sennrich et al., 2016) as a baseline. Moreover, we have used the methods that have obtained the best results with languages of different types of morphology. We used Unigram (Kudo, 2018) because it is considered the best unsupervised and statistically motivated method, as it has obtained interesting results for both morphologically complex languages (e.g. Quechua) and non-complex languages (e.g. English) (Gow-Smith et al., 2022). In the case of canonical and linguistically motivated methods, we chose DeepSpin (Peters and Martins, 2022), which is the winner of SIGMORPHON 2022 (Batsuren et al., 2022), achieving very interesting results and superior to others of the same type of tokenization.

Because DeepSpin is a supervised model, it is necessary to train a model for each language. The data used to train the English and French model were obtained from SIGMORPHON[3] 2022 itself, the German data from the experiments performed by Cotterell et al. (2016). The Turkish and Quechua training data were created by ourselves for these

[2]Dataset from https://huggingface.co/datasets/Llamacha/monolingual-quechua-iic

[3]Dataset from https://github.com/sigmorphon/2022SegmentationST/tree/main/data

experiments. The Turkish raw data was obtained from Alecakir et al. (2022), and the Quechua raw data was obtained from Melgarejo et al. (2022). All models were trained with the same hyperparameters as DeepSpin-Base (Peters and Martins, 2022). In Table 2, we show the data obtained from the trained DeepSpin models of each language.

Language	Annotated words	Accuracy
English	458k	0.92
French	382k	0.94
German	8k	0.83
Turkish	2k	0.75
Quechua	1k	0.72

Table 2: Annotated dataset size and DeepSpin tokenization accuracy were considered in this study. For each language, DeepSpin was trained using an 80/10/10 split for training, validation, and testing, respectively.

3.1.3 Hyperparameters

To replicate what Warstadt et al. (2020) did for data smaller than 10M tokens, we used the hyperparameters from their Med-Small model, which had 8 attention heads, 512 hidden size, 2048 feed-forward network dimension, and 45M parameters. Note that we have also set the vocabulary size to 52,000 tokens just like most experiments in language model development with transformers. This size of 52k tokens is due to a computational limitation when processing the data. In addition, we adopted the same parameter values for dropout, attention dropout and learning rate decrease. All parameters are described in Table 3.

Description	**Value**
Number of attention heads	8
Hidden size	512
Feed-forward network dimension	2048
Number of parameters	45M
Max Steps	10K
Batch Size	512
Dropout	0.1
Attention dropout	0.1
Learning rate decrease	5E-4

Table 3: Common parameters for the pretraining of the 20 models used in our experiments.

3.2 Fine-Tuning

From the pretrained RoBERTa models, and still following Zhang et al. (2021), we generated representations of the token span and trained classifiers that predict whether a given label correctly describes the input span for NER and POS.

In order to obtain the best and validated results in both tasks, we performed a 10-fold macro-F1 score cross-validation. In addition, we chose to adjust some hyperparameters guided by Zhang et al. (2021): learning rate $\in$ {1E-5, 2E-5, 3E-5, 4E-5} and batch size $\in$ {16, 32, 48}.

In POS tagging, we used a different head with a classification output for each token, all triggered by a softmax function just like Delobelle et al. (2020). Also, when a word consists of multiple tokens, the first token is used for the word tag. The xtreme[4] (Conneau et al., 2018) datasets were used for the POS task and wikiann[5] (Rahimi et al., 2019) for the NER of English, German, French, and Turkish. For Quechua[6], the dataset provided by Zevallos et al. (2022) was used for both tasks. For evaluating the NER and POS tasks, we used macro-F1 score.

4 Results

Our research aimed on the one hand at making an evaluation of the amount of pretraining data and the role of the tokenizer measured in terms of LM perplexity. On the other hand, the POS and NER tasks were meant to assess the quality of the representations produced when used in fine-tuning downstream tasks. It is important to mention that we did not perform any normalization in the results as opposed to Warstadt et al. (2020), because we also wanted to see a comparison between languages.

4.1 Pretrained models

The results per language plotted in Figure 1 show that for all cases the DeepSpin tokenization method substantially improves the perplexity of all LM, but it is in the case of Turkish and Quechua that it drastically improves the perplexity from 162.47 to 94.93 and 210.14 to 102.73 respectively. English LM obtained 53.51, being the lowest perplexity in all the configurations performed in the experiments. Comparing BPE and Unigram, only English, German and French achieved better results, while Turkish and Quechua also achieved better results using Unigram.

We can see that the datasize amounts that are critical for modeling English (Warstadt et al., 2020)

[4] https://huggingface.co/datasets/xtreme
[5] https://huggingface.co/datasets/wikiann
[6] https://github.com/Llamacha/QuBERT/tree/main/resource

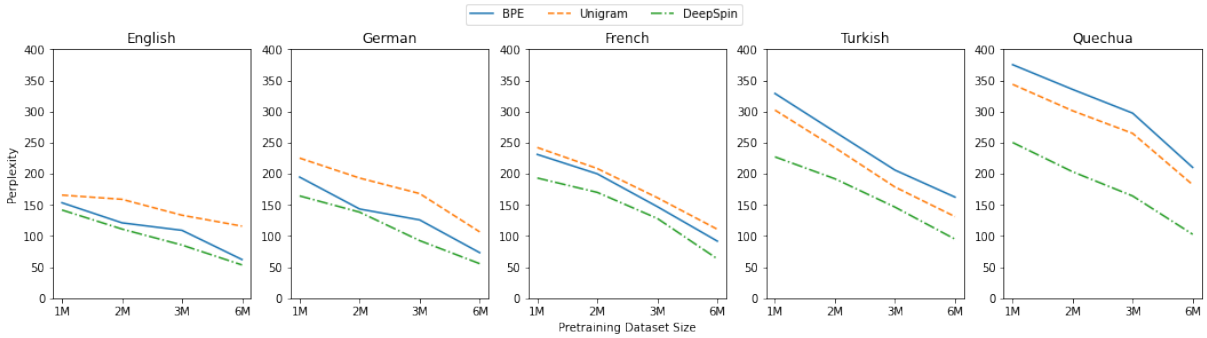

Figure 1: Perplexity for each language model, training data size (1M, 2M, 3M and 6M) and tokenizer. Numeric data can be found in the Appendix.

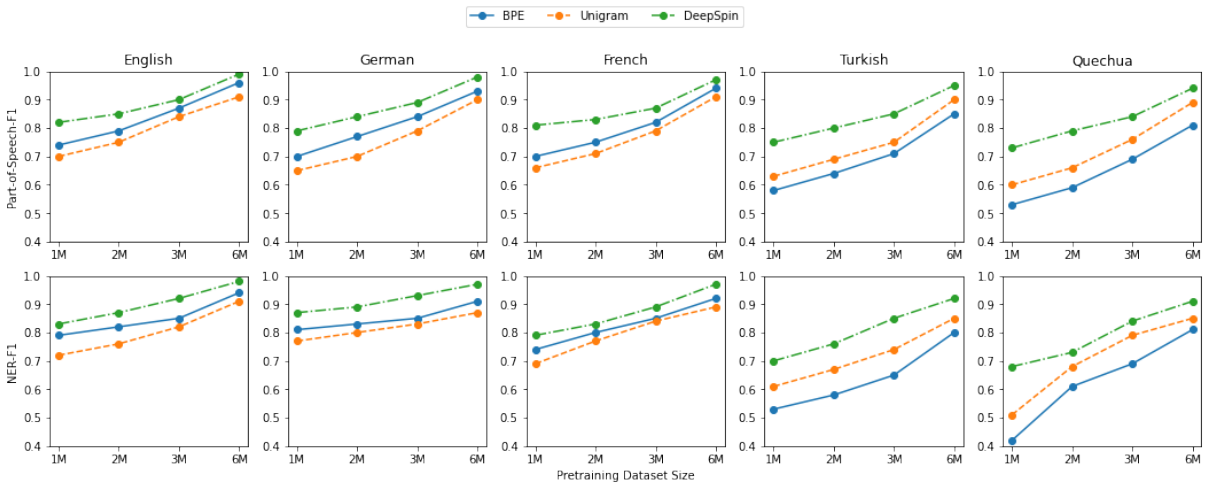

Figure 2: Macro-F1 score NER and POS results for each language and data sizes (1M, 2M, 3M and 6M) of the pretrained models.

are quite different for other languages. Despite having the same training data size and using the same training hyperparameters and vocabulary limitations, the results in terms of LM perplexity are very different. The perplexity of the Turkish and Quechua language models is around twice the perplexity of the English LM with 6M with all the tokenizers. In the appendix we show all the results of the pretrained models according to type of tokenization.

4.2 Part-of-Speech Tagging

We evaluated POS tagging task for each language with the different training sizes and different tokenization methods. For all models, the same hyperparameters mentioned in 3.2 are used. In Table 4 and Figure 2 we can see the results for all dataset sizes for each language and the different tokenization methods.

Zhang et al. (2021) found that POS labelling was one of the taks whose learning curve rises earlier and gets around 90% macro-F1 score with less than 10M of training dataset. Our results, in Table 4, show the same trend, with English, German and French getting macro-F1 score higher than 90% with a corpus of 6M and the three tokenizers. For Turkish and Quechua, BPE tokenization is the only one that cannot achieve a 90% macro-F1 score.

As can be seen in the Table 4, for all languages and the 6M dataset, using the DeepSpin tokenizer delivers statistically significant improvements[7] both when compared to BPE, that works better for English, German and French, and when compared to Unigram, that, as expected, works better for Turkish and Quechua. What is more interesting is that for Turkish and Quechua with DeepSpin better results are obtained with 3M words than BPE with 6M, showing the importance of the tokenizer selection for synthetic languages.

[7] Sign test, $p < 0.05$

Language	BPE				Unigram				DeepSpin			
	1M	2M	3M	6M	1M	2M	3M	6M	1M	2M	3M	6M
English	0.74	0.79	0.87	0.96	0.70	0.75	0.84	0.91	0.82	0.85	0.90	0.99
German	0.70	0.77	0.84	0.93	0.65	0.70	0.79	0.90	0.79	0.84	0.89	0.98
French	0.70	0.75	0.82	0.94	0.66	0.71	0.79	0.91	0.81	0.83	0.87	0.97
Turkish	0.58	0.64	0.71	0.85	0.63	0.69	0.75	0.90	0.75	0.80	0.85	0.95
Quechua	0.53	0.59	0.69	0.81	0.60	0.66	0.76	0.89	0.73	0.79	0.84	0.94

Table 4: Macro-F1 score results of the POS tagging task for each language, using the subset of 1M, 2M, 3M and 6M words and three different tokenization methods.

Language	BPE				Unigram				DeepSpin			
	1M	2M	3M	6M	1M	2M	3M	6M	1M	2M	3M	6M
English	0.79	0.82	0.85	0.94	0.72	0.76	0.82	0.91	0.83	0.87	0.92	0.98
German	0.81	0.83	0.85	0.91	0.77	0.80	0.83	0.87	0.87	0.89	0.93	0.97
French	0.74	0.80	0.85	0.92	0.69	0.77	0.84	0.89	0.79	0.83	0.89	0.97
Turkish	0.53	0.58	0.65	0.80	0.61	0.67	0.74	0.85	0.70	0.76	0.85	0.92
Quechua	0.42	0.61	0.69	0.81	0.51	0.68	0.79	0.85	0.68	0.73	0.84	0.91

Table 5: Macro-F1 score results of the NER task for each language, using the subset of 1M, 2M, 3M and 6M words and three different tokenization methods.

4.3 Named Entity Recognition

In Figure 2 we can see the results for all dataset sizes for each language and the different tokenization methods (figures can be found in Table 5). For NER tasks, Zhang et al. (2021) results showed that the learning curve still raised between 10M and 100M datasets before stalling. Our results show that the learning curve for NER is sharper than for POS tagging: it needs more data for all languages, but again Turkish and Quechua having more difficulties in all cases. However, when using the DeepSpin tokenizer, statistically significant improvements are achieved for each language with all datasizes. In the case of Turkish and Quechua, DeepSpin achieves the same macro-F1 score results than Unigram with the 3M dataset, and improves BPE results with the 6M dataset.

5 Discussion

In order to clarify the amount of data necessary to achieve robust performance measured by LM perplexity, we experimented with four training data sizes: 1M, 2M, 3M and 6M tokens. We were interested in two main issues. First, in the work of Warstadt et al. (2020) it can be seen that perplexity improves dramatically when the training data size is above 10M, however low-resource languages like Quechua do not even have texts amounting 10M tokens. We were interested in finding whether there is a critical amount of data with which it is worth for low-resource languages to build a TLM. Second, we wanted to show to what extent LM perplexity and the fine-tuning of downstream tasks are influenced by the size of the data and the morphological typology of languages, and whether tokenization could mitigate these issues.

From our results it is clear that in spite of being trained with the same configurations and amount of training data, there are differences among the languages we examined. Mielke et al. (2019) suggested that these differences could be due to the difference in morphological complexity between these languages. A rich inflectional morphology increases the vocabulary. As we can see in Table 6, tokenizers that try to identify the compositional characteristics of morphology can significantly reduce the vocabulary size. Therefore, the drastic improvement in the perplexity results for Quechua, with perplexity 210 with BPE and 102 with DeepSpin, is due to the fact that DeepSpin manages to reduce the vocabulary thanks to a linguistically motivated segmentation.

We also wanted to get evidence about the quality of the representations obtained by our different TLM for fine-tuning downstream tasks. The results shown in Table 4 and Table 5 show that representations get better with more data, but a TLM trained with dataset of 6M tokens and a using a linguistically motivated tokenizer can deliver very

Language	BPE				Unigram				DeepSpin			
	1M		6M		1M		6M		1M		6M	
	Voc.	TTR	Voc.	TTR	Voc.	TTR	Voc.	TTR	Voc.	TTR	Voc.	TTR
English	20.3	0.203	51	0.084	30.1	0.301	51.3	0.085	14.1	0.141	16.2	0.027
French	20.9	0.209	52	0.085	30.7	0.307	51.6	0.086	14.2	0.142	22.8	0.038
German	21.5	0.215	52	0.085	31.4	0.314	51.6	0.086	14.7	0.147	25.2	0.042
Turkish	21.9	0.219	52	0.086	32.1	0.321	52	0.086	15.1	0.151	28.2	0.047
Quechua	22.1	0.221	52	0.086	33.4	0.334	52	0.086	15.3	0.153	32.2	0.053

Table 6: Vocabulary size (Voc.) and Type-Token Ratio (TTR) of each language according to the size of the training data and the tokenization method. TTR is multiplied by 10^3 and Voc. is divided by 10 to better appreciate the results.

competitive results for tasks like POS tagging and NER.

6 Conclusions

In this paper we have related the quality of TLM with the training data size. We have approached the topic from the point of view of low-resource languages that need to maximize the available data. We have demonstrated how different methods, in this case tokenizers, apply to languages other than English. We have evaluated intrinsically and extrinsically the impact of datasize and tokenization with the aim of giving some hints for the building of TLM for low-resource languages, in particular for those whose morphology processes produces large vocabularies. These hints are explaining below.

6.1 How much data is enough?

In our experiments, all languages show a continuous reduction of perplexity when from 1M to 6M tokens, with no stagnation. Regardless of language type, the decrease in perplexity progresses as the model is trained with more data, suggesting that it can still improve more with more data. However, we provide evidence on the fact that with 6M all the languages in our experiments, but Turkish and Quechua, could reach a perplexity below 100, and macro-F1 score higher than 0.9 in the two downstream tasks. With a linguistically motivated and canonical tokenizer like DeepSpin, Turkish and Quechua could also attain these competitive results, as explained below.

6.2 Which tokenizer to use?

Tokenization methods play an important role in building pretrained models (Rust et al., 2021). As seen in our experiments, canonical and linguistically motivated tokenizers achieve astonishing results compared to other types of tokenizers. The reduction by almost 50% of the perplexity of the pre-entangled models of Turkish and Quechua when using DeepSpin instead of BPE is impressive. Languages morphologically different from Turkish and Quechua also showed significant benefits, e.g., English, French and German showed an improvement of 15%, 31% and 24% respectively.

On the other hand, it can also be seen that using DeepSpin results in significant improvements in tasks such as NER and POS tagging. Both Turkish and Quechua manage to increase the macro-F1 score by 0.1 and 0.14 respectively. English, French and German also manage to increase the macro-F1 score by 0.03 in most cases.

Finally, we can say that canonical and linguistically motivated tokenization methods present statistically significant improvements when working with morphologically complex languages compared to statistically motivated methods such as BPE and Unigram.

7 Limitations

We have limited ourselves to experimenting with only five languages due to lack of data for both the pretrained models and the DeepSpin tokenizer models. Although there are annotated data for some low-resource polysynthetic languages such as Nahuatl, Raramuri, Wixarika, Shipibo-Konibo (Mager et al., 2020) and Kunwinjku (Pimentel et al., 2021), the available data was below 1M and therefore not enough to create pretrained models for our experiments.

Regarding the aforementioned limitation, DeepSpin which has proven to be a good option to mitigate the problem of high TTR languages in closed vocabulary environments is a supervised method that requires the availability of training data. As

can be seen in Table 2, to achieve 90% to better accuracy DeepSpin requires around 350K annotated words. This can be a major drawback for low-resource languages, although the results with less annotated data are still competitive. We have not studied another source of differences in the vocabulary size that could be due to the texts used in pretraining. Ortiz Suárez et al. (2019) found that, in general, the OSCAR samples contain more vocabulary words than the Wikipedia ones. Additionally, the Quechua corpus we have used also consists of educational and legal texts that can increase the number of different types, compared to Wikipedia texts.

On the other hand, we believe it is important to mention that for the Quechua language the training, evaluation, and testing data for NER and POS tasks were obtained from the same corpus used for training the language model. Note that, due to the scarcity of available digital and physical texts in that language, it is difficult to do it otherwise. The limited availability of texts leads to the use of the same corpus for multiple tasks, which could have implications on the evaluation of the obtained results. For instance, if the training corpus contains an unequal proportion of certain types of grammatical structures, it might negatively affect the performance of POS classifiers. Furthermore, if the corpus does not adequately reflect the linguistic variability and diversity of Quechua, the resulting models are likely to be less accurate and less generalizable.

Ethical Considerations

The datasets used in this paper for the training and evaluations of the pre-trained models, DeepSpin models, and fine-tuned models have been extracted from various previous articles and open-access repositories, therefore, we abide by the ethical rules by citing the original authors of each dataset. On the other hand, the annotated Turkish and Quechua data that were constructed by us for the development of the DeepSpin models will be presented in a forthcoming paper for public use. In addition, we encourage authors who use the resources in this article to cite the original sources. Finally, we would like to note that one of the authors of this paper has a long history of working with resource-poor synthetic languages, especially Quechua, which allows us to better understand the problems and concerns of the Quechua-speaking communities.

Acknowledgements

This research was partially funded by the project LUTEST, Project PID2019-104512GB-I00, Ministerio de Ciencia, Innovación y Universidades and Agencia Estatal de Investigación (Spain). The first author has been supported by a FI grant of the Catalan Funding Agency for Research and Universities (AGAUR).

References

Huseyin Alecakir, Necva Bölücü, and Burcu Can. 2022. TurkishDelightNLP: A neural Turkish NLP toolkit. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: System Demonstrations*, pages 17–26, Hybrid: Seattle, Washington + Online. Association for Computational Linguistics.

Rachit Bansal, Himanshu Choudhary, Ravneet Punia, Niko Schenk, Jacob L. Dahl, and Émilie Pagé-Perron. 2021. How low is too low? A computational perspective on extremely low-resource languages. *CoRR*, abs/2105.14515.

Khuyagbaatar Batsuren, Gábor Bella, Aryaman Arora, Viktor Martinovic, Kyle Gorman, Zdeněk Žabokrtský, Amarsanaa Ganbold, Šárka Dohnalová, Magda Ševčíková, Kateřina Pelegrinová, Fausto Giunchiglia, Ryan Cotterell, and Ekaterina Vylomova. 2022. The SIGMORPHON 2022 shared task on morpheme segmentation. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 103–116, Seattle, Washington. Association for Computational Linguistics.

William Chen and Brett Fazio. 2021. Morphologically-guided segmentation for translation of agglutinative low-resource languages. In *Proceedings of the 4th Workshop on Technologies for MT of Low Resource Languages (LoResMT2021)*, pages 20–31, Virtual. Association for Machine Translation in the Americas.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485.

Ryan Cotterell, Tim Vieira, and Hinrich Schütze. 2016. A joint model of orthography and morphological segmentation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 664–669.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. Robbert: a Dutch roberta-based language

model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3255–3265.

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.

P. Geutner. 1995. Using morphology towards better large-vocabulary speech recognition systems. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 445–448 vol.1.

Edward Gow-Smith, Harish Tayyar Madabushi, Carolina Scarton, and Aline Villavicencio. 2022. Improving tokenisation by alternative treatment of spaces. *arXiv preprint arXiv:2204.04058*.

Naman Goyal, Jingfei Du, Myle Ott, Giri Anantharaman, and Alexis Conneau. 2021. Larger-scale transformers for multilingual masked language modeling. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 29–33.

Jennifer Hu, Jon Gauthier, Peng Qian, Ethan Wilcox, and Roger Levy. 2020. A systematic assessment of syntactic generalization in neural language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1725–1744, Online. Association for Computational Linguistics.

Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 2021. The interplay of variant, size, and task type in Arabic pre-trained language models. In *Workshop on Arabic Natural Language Processing*.

Kimmo Kettunen. 2014. Can type-token ratio be used to show morphological complexity of languages? *Journal of Quantitative Linguistics*, 21:223–245.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 66–71. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Manuel Mager, Özlem Çetinoğlu, and Katharina Kann. 2020. Tackling the low-resource challenge for canonical segmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5237–5250, Online. Association for Computational Linguistics.

Manuel Mager, Arturo Oncevay, Elisabeth Mager, Katharina Kann, and Thang Vu. 2022. BPE vs. morphological segmentation: A case study on machine translation of four polysynthetic languages. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 961–971, Dublin, Ireland. Association for Computational Linguistics.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online. Association for Computational Linguistics.

Nelsi Melgarejo, Rodolfo Zevallos, Hector Gomez, and John E. Ortega. 2022. WordNet-QU: Development of a lexical database for Quechua varieties. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4429–4433, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Kurt Micallef, Albert Gatt, Marc Tanti, Lonneke van der Plas, and Claudia Borg. 2022. Pre-training data quality and quantity for a low-resource language: New corpus and BERT models for Maltese. In *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*, pages 90–101, Hybrid. Association for Computational Linguistics.

Vincent Micheli, Martin d'Hoffschmidt, and François Fleuret. 2020. On the importance of pre-training data volume for compact language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7853–7858, Online. Association for Computational Linguistics.

Sabrina J. Mielke, Ryan Cotterell, Kyle Gorman, Brian Roark, and Jason Eisner. 2019. What kind of language is hard to language-model? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4975–4989, Florence, Italy. Association for Computational Linguistics.

John E. Ortega, Richard Castro Mamani, and Kyunghyun Cho. 2020. Neural machine translation with a polysynthetic low resource language. *Machine Translation*, 34(4):325–346.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low-resource infrastructures. Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

Hyunji Hayley Park, Katherine J. Zhang, Coleman Haley, Kenneth Steimel, Han Liu, and Lane Schwartz. 2021. Morphology matters: A multilingual language modeling analysis. *Transactions of the Association for Computational Linguistics*, 9:261–276.

Laura Pérez-Mayos, Miguel Ballesteros, and Leo Wanner. 2021. How much pretraining data do language models need to learn syntax? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1571–1582, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ben Peters and Andre F. T. Martins. 2022. Beyond characters: Subword-level morpheme segmentation. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 131–138, Seattle, Washington. Association for Computational Linguistics.

Tiago Pimentel, Maria Ryskina, Sabrina J. Mielke, Shijie Wu, Eleanor Chodroff, Brian Leonard, Garrett Nicolai, Yustinus Ghanggo Ate, Salam Khalifa, Nizar Habash, Charbel El-Khaissi, Omer Goldman, Michael Gasser, William Lane, Matt Coler, Arturo Oncevay, Jaime Rafael Montoya Samame, Gema Celeste Silva Villegas, Adam Ek, Jean-Philippe Bernardy, Andrey Shcherbakov, Aziyana Bayyr-ool, Karina Sheifer, Sofya Ganieva, Matvey Plugaryov, Elena Klyachko, Ali Salehi, Andrew Krizhanovsky, Natalia Krizhanovsky, Clara Vania, Sardana Ivanova, Aelita Salchak, Christopher Straughn, Zoey Liu, Jonathan North Washington, Duygu Ataman, Witold Kieraś, Marcin Woliński, Totok Suhardijanto, Niklas Stoehr, Zahroh Nuriah, Shyam Ratan, Francis M. Tyers, Edoardo M. Ponti, Grant Aiton, Richard J. Hatcher, Emily Prud'hommeaux, Ritesh Kumar, Mans Hulden, Botond Barta, Dorina Lakatos, Gábor Szolnok, Judit Ács, Mohit Raj, David Yarowsky, Ryan Cotterell, Ben Ambridge, and Ekaterina Vylomova. 2021. SIGMORPHON 2021 shared task on morphological reinflection: Generalization across languages. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 229–259, Online. Association for Computational Linguistics.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for ner. *arXiv preprint arXiv:1902.00193*.

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. How good is your tokenizer? on the monolingual performance of multilingual language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.

Raphael Scheible, Fabian Thomczyk, Patric Tippmann, Victor Jaravine, and Martin Boeker. 2020. Gottbert: a pure German language model. *arXiv preprint arXiv:2012.02110*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Cagri Toraman, Eyup Halit Yilmaz, Furkan Şahinuç, and Oguzhan Ozcelik. 2022. Impact of tokenization on language models: An analysis for Turkish. *arXiv preprint arXiv:2204.08832*.

Alex Warstadt, Yian Zhang, Xiaocheng Li, Haokun Liu, and Samuel R. Bowman. 2020. Learning which features matter: RoBERTa acquires a preference for linguistic generalizations (eventually). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 217–235, Online. Association for Computational Linguistics.

Edward Whittaker and Philip Woodland. 2003. Language modelling for Russian and English using words and classes [computer speech and language 17 (2003) 87–104]. *Computer Speech & Language*, 17:415.

Rodolfo Zevallos, John Ortega, William Chen, Richard Castro, Núria Bel, Cesar Toshio, Renzo Venturas, Hilario Aradiel, and Nelsi Melgarejo. 2022. Introducing QuBERT: A large monolingual corpus and BERT model for Southern Quechua. In *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*, pages 1–13, Hybrid. Association for Computational Linguistics.

Yian Zhang, Alex Warstadt, Xiaocheng Li, and Samuel Bowman. 2021. When do you need billions of words of pretraining data? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1112–1125.

A Appendices

A.1 Model and training procedure: details

To train language models for each language, we followed the choices by Warstadt et al. (2020) for their RoBERTa Med-Small model with 45M parameters, based on the amount of training data (<10M).

We ran all training in parallel on five servers, with each language on a separate server. All servers were equipped with an Intel Xeon E5-2650 v4 CPU (12 cores, 2.2GHz 30MB Cache 2400MHz 105W)

and a Gigabyte Geforce GTX 1080 Ti TURBO 11.72GB GPU. We trained each model for 10k steps, and the training time varied depending on the amount of training data. The models trained on 1M, 2M, 3M, and 6M took 16 hours, 1 day, 2 days, and 3 days, respectively. The entire LM creation experiment took approximately 7 days. Fine-tuning the POS and NER models for Quechua took 2 days; for Turkish, it took 4 days; for French and German took 5 days each, and for English, it took 10 days. We performed each fine-tuning process using 1k steps and each fine-tuning process was carried out on the same server that was used to train the language model.

A.2 Experiment results

The following Tables show the perplexity of the different language models trained with different tokenization methods and amount of training tokens. Table 7 shows perplexity of the language models that used Unigraman as a tokenization method, while Table 8 shows perplexity with BPE and Table 9 with DeepSpin.

Language	Tokens (Millions)	Perplexity	Language	Tokens (Millions)	Perplexity
English		153.38	English		109.07
German		194.62	German		125.83
French	1	231.03	French	3	147.15
Turkish		328.91	Turkish		205.83
Quechua		375.17	Quechua		297.29
English		121.14	English		62.15
German		143.33	German		73.21
French	2	199.80	French	6	91.72
Turkish		267.22	Turkish		162.47
Quechua		335.41	Quechua		210.14

Table 7: Perplexity for each language and training data size using the BPE tokenization method.

Language	Tokens (Millions)	Perplexity	Language	Tokens (Millions)	Perplexity
English		165.72	English		133.37
German		225.13	German		168.11
French	1	242.15	French	3	161.5
Turkish		302.24	Turkish		178.61
Quechua		343.61	Quechua		264.82
English		158.91	English		115.82
German		193.06	German		106.62
French	2	208.35	French	6	110.80
Turkish		241.77	Turkish		131.09
Quechua		301.09	Quechua		182.35

Table 8: Perplexity for each language and training data size using the Unigram tokenization method.

Language	Tokens (Millions)	Perplexity	Language	Tokens (Millions)	Perplexity
English		141.77	English		85.42
German		164.39	German		92.61
French	1	193.16	French	3	128.19
Turkish		227.11	Turkish		146.38
Quechua		250.18	Quechua		164.25
English		111.13	English		53.51
German		138.28	German		55.53
French	2	170.03	French	6	63.28
Turkish		191.88	Turkish		94.93
Quechua		203.15	Quechua		102.73

Table 9: Perplexity for each language and training data size using the DeepSpin tokenization method.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
7

☒ A2. Did you discuss any potential risks of your work?
We have not created any system or any dataset that can have a potential missuse.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and 1. Introduction

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

3

☑ B1. Did you cite the creators of artifacts you used?
3

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Left blank.

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
We only used very common systems whose inteded use is well-known and we have used them in a standard way.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
3

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
3

C ☑ Did you run computational experiments?

3

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
3

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
We mention the search parameters in section 3 and appendices

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
4

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
3

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?
3

☒ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
The annotation was not the focus of the paper.

☒ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
The annotation was not the focus of the paper.

☒ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
The annotation was not the focus of the paper.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.