The Impact of Data Corruption on Named Entity Recognition for Low-resourced Languages

Abstract

Data availability and quality are major challenges in natural language processing for low-resourced languages. In particular, there is significantly less data available than for higher-resourced languages. This data is also often of low quality, rife with errors, invalid text or incorrect annotations. Many prior works focus on dealing with these problems, either by generating synthetic data, or filtering out low-quality parts of datasets. We instead investigate these factors more deeply, by systematically measuring the effect of data quantity and quality on the performance of pre-trained language models in a low-resourced setting. Our results show that having fewer completely-labelled sentences is significantly better than having more sentences with missing labels; and that models can perform remarkably well with only 10% of the training data. Importantly, these results are consistent across ten low-resource languages, English, and four pre-trained models.

1 Introduction

Natural Language Processing (NLP) is a rapidly growing field that has been applied to a wide range of tasks and domains [Vaswani *et al.*, 2017; Conneau *et al.*, 2020]. However, much of the focus in NLP has been on high-resource languages such as English [Vaswani *et al.*, 2017; Radford *et al.*, 2018, 2019]. While this has led to notable advancements for these languages, low-resourced languages have not received as much attention, resulting in a significant performance gap between high- and low-resourced languages. This has prompted an increasing number of studies focused exclusively on low-resourced languages, resulting in the development of models [Ogueji *et al.*, 2021; Alabi *et al.*, 2022] and the introduction of datasets [Adelani *et al.*, 2021, 2022a,b].

Despite this impressive progress, data remains a limiting factor for low-resourced NLP [Adelani *et al.*, 2022a,b]. In particular, the two main problems are the availability and quality of data. First, the datasets available for low-resourced languages tend to be smaller than those for high-resourced languages, and for many languages, no data exists at all [Martinus and Abbott, 2019; Adelani *et al.*, 2021]. Secondly, the available datasets are often of questionable quality, containing invalid text or incorrect annotations [Kreutzer *et al.*, 2022], which has detrimental effects on the models trained on these datasets [Abdul-Rauf *et al.*, 2012; Alabi *et al.*, 2019].

This means that many existing datasets in low-resourced NLP are either small or of low quality. This observation has led to research that investigates the tradeoff between the amount and quality of data [Gascó *et al.*, 2012; Alabi *et al.*, 2019; de Gibert Bonet *et al.*, 2022]. This line of work has provided valuable insights that allow NLP practitioners to make informed decisions when faced with a choice of which dataset should be used to train a model. However, many of these works focus on comparing different datasets, often from different domains, without clearly quantifiable tradeoffs between data quantity and quality [Alabi *et al.*, 2019]. While these approaches can be useful, a more comprehensive and precise approach is needed to fully understand the tradeoff between data quantity and quality in low-resourced NLP.

To address this, we take a different perspective and focus on systematically and quantifiably reducing the quality of datasets and examining the effects of this on the performance of NLP models. Additionally, by altering the amount of data used to train our models, we can compare the tradeoffs between quality and quantity. We do this by devising various controllable corruption strategies, and training models on different levels of corrupted data. Our focus is on a named-entity recognition task due to its prevalence in many NLP systems and the availability of a few high-quality datasets in low-resourced languages. We fine-tune existing pre-trained language models, as this is a common and high-performing approach, especially for low-resourced languages [Ogueji *et al.*, 2021; Adelani *et al.*, 2021; Alabi *et al.*, 2022].

We provide systematic evidence to support prior findings that the quality of data, in general, is strongly preferred over quantity. Furthermore, our findings are consistent across eleven different languages and four pre-trained models, suggesting that our conclusions hold true in a general sense.

2 Background and Related Work

2.1 Named Entity Recognition

Named entity recognition (NER) is a token classification task, where the goal is to classify each token or word in a text as an Organisation, Location, Person, Date, or indicate that the

token does not correspond to a named entity by giving it the label of "Other". NER as a field has many impactful applications in NLP pipelines and use-cases[Sang and Meulder, 2003; Lample and Chaplot, 2017; Adelani *et al.*, 2021]. A typical NER dataset consists of multiple sentences, with each sentence containing both the words and their associated labels.

The prevailing approach to train NER models is to use a pre-trained large language model (such as BERT [Devlin *et al.*, 2019], XLM-Roberta [Conneau *et al.*, 2020], etc.) and fine-tune it on a small amount of NER data [Conneau *et al.*, 2020; Adelani *et al.*, 2021]. These models were pre-trained on a large corpus of unlabelled text, and resulted in improved downstream performance after fine-tuning compared to training on NER data from scratch. The overall classification F1 score, calculated as the harmonic mean of precision and recall, is generally used as the main metric of performance in NER [Sang and Meulder, 2003; Adelani *et al.*, 2021].

2.2 Data Collection and Annotation

Since the lack of data has traditionally been a major limiting factor for low-resourced NLP research, multiple different approaches have developed to effectively collect data in resource-constrained settings. In particular, community involvement has played a large part in this [Nekoto *et al.*, 2020, 2022], where native speakers annotate or create datasets to be used in research. This has led to the creation of many different datasets [Adelani *et al.*, 2021; Nekoto *et al.*, 2022], but it relies on community members instead of trained annotators, which may result in some aspects of the annotation being less accurate. Furthermore, while this approach can successfully develop datasets for low-resourced languages, due to logistic challenges and a limited amount of unlabelled text, these datasets are often significantly smaller than high-resourced datasets [Conneau *et al.*, 2020; Adelani *et al.*, 2021].

2.3 Analysis of Quality vs Quantity in Low-resourced Languages

While there has been significant progress in recent years, datasets for low-resourced language are often quite small and limited, or exhibit low quality. Both of these factors can lead to poorly-performing models. For instance, Kreutzer *et al.* [2022] perform a large-scale audit of several web-scale and automatically extracted multilingual datasets, and find that the quality is often poor, with non-linguistic or otherwise invalid text being commonplace.

This lack of quality can have great effects on the performance of models. Alabi *et al.* [2019] show that for certain low-resourced African languages, using a significantly smaller, but curated dataset outperforms training a model on a large, but noisy dataset. Abdulmumin *et al.* [2022] find similar results, where training on filtered data of higher quality improved the performance of translation models for low-resourced languages. Many of these works consider one or two completely different datasets, and compare the relative quality and quantity. This, however, lacks a systematic approach that controls for other factors such as the domain of the data. In addition, the filtering-based approaches often use a learned model as a filter and select only sentences that have a predicted quality value above a certain threshold [Abdulmumin *et al.*, 2022; de Gibert Bonet *et al.*, 2022]. While this does provide a quantifiable level of quality, it may not be comparable across datasets or different filtering models. Additionally, datasets may exhibit different levels of certain problems, e.g. some datasets may have many tokens corresponding to punctuation whereas others may have sentences in a different language to the rest of the data. These problems may make it hard to accurately compare the results of these studies and use their conclusions in practice.

3 Methodology

Our aim is to analyse and quantify the impact of data corruption on the performance of pre-trained language models. This understanding would enable us to make more informed decisions about the relative importance of data quality and quantity, ultimately leading to improved data creation processes and selection of NLP training data for practitioners.

While we can corrupt NER datasets in various ways, we choose corruptions that simulate a mislabelling scenario during the annotation process, e.g. mislabelling a person in a sentence as an organisation. There are two main reasons for this choice. Firstly, many NER datasets are formed by taking an existing text source, which is usually of high quality, such as news data [Adelani *et al.*, 2021] and annotating each word; thus, errors are more likely to appear during the annotation process. Secondly, it is challenging to corrupt the base sentences in a reasonable, quantifiable and incremental way, as sentences encompass meaning which is often hard to change atomically.

Thus, we focus on corrupting only the labels, using different strategies detailed in Section 3.1. For each corruption strategy, we uniformly vary the amount of corruption and train our models using the new, corrupted dataset. This process allows us to evaluate how each corruption strategy affects the model as we adjust the degree of corruption. As an additional experiment, we vary the size of the data available to the model by using only a subset of the sentences without corrupting any labels, allowing us to determine the effect of varying the amount of data on performance.

We discuss the data used in this study in Section 3.2.

3.1 Different Corruption Strategies

This section contains descriptions of the corruption strategies that we use, with Figure 1 providing a visual illustration of our quality-related corruption strategies. We only change the training data while leaving the evaluation data unchanged to obtain an objective comparison of different corruption strategies.

Sentence Capping

Dataset annotation is generally expensive and logistically challenging when multiple participants are involved. As a result, low-resourced NLP datasets are often not particularly large [Adelani *et al.*, 2021]. Due to this observation, we first evaluate the effect of varying the size of the data available to our NER models. In this strategy, we randomly remove sentences from the original dataset to create sub-datasets with fewer sentences than the original dataset. This process allows

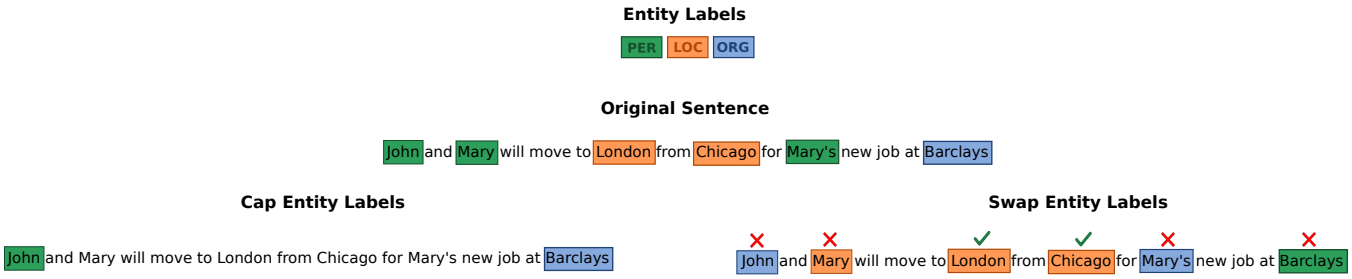


Figure 1: An illustration of the different corruption strategies we use. When (Left) capping labels, we effectively remove a certain fraction of labels, replacing them with *O*. When (Right) swapping labels, we instead randomly replace a label with an incorrect one. This figure is just illustrative, and in our experiments we have a percentage threshold; for instance, corrupting 30% of the labels across the entire dataset.

us to measure the model performance as a function of data quantity. We choose to represent quantity as a function of the number of sentences because removing words can alter the meaning of a sentence in ways we cannot control.

Entity Label Capping

A rich NER dataset would be a dataset with a high annotation density, i.e. a high number of annotated entities per sentence. This strategy aims at inhibiting the model by thresholding the number of entity annotations allowed in the dataset. In the real world, this would be equivalent to a situation where an annotator failed to label a particular token or span of tokens as one of the entities PER, LOC, ORG, DATE, instead giving it the default entity type *O*, which generally means *not relevant*.

Here we globally corrupt the data by choosing a certain percentage of labels to keep across the entire dataset. For example, 50% would mean that we randomly remove half of all entity labels (replacing them with *O*), which may leave some sentences unmodified and others entirely without annotations.

For this corruption strategy and the next one, we consider the atomic element to be a single named entity, even if this consists of multiple words. As a result, we change the entire span of an entity label instead of just a part thereof.

Entity Label Swapping

Another scenario that could happen during the annotation procedure would be the mislabelling of a span of tokens with the wrong entity. For example, A person named *Christian Dior* mistakenly labelled as an organisation due to some bias in the knowledge of the annotator. These mistakes may create datasets with contradictory labels, with the same tokens being used in very similar contexts but labelled differently. Therefore, our goal behind this corruption strategy is to determine how robust large pre-trained language models are to such mistakes. Here we again choose a global percentage, randomly selecting labelled entities according to this percentage and swapping their labels with incorrect ones.

3.2 Data

We use the MasakhaNER dataset [Adelani *et al.*, 2021], a high-quality NER dataset for ten low-resourced African languages. We specifically focus on low-resourced languages, as these languages often suffer from the aforementioned problems. Furthermore, this dataset is of high quality, which allows us to evaluate the full spectrum of quality, from gold-standard to completely corrupted. As a baseline, we also use the CONLL NER dataset, which is a staple NER dataset in English [Sang and Meulder, 2003], with many more sentences than any of the MasakhaNER languages. Table 1 contains information about the number of sentences and entities for each NER corpus.

Table 1: Information about the data. Entity Density refers to the fraction of tokens that are entities. All languages use the Latin script, except Amharic, which uses the Fidel script.

Code	Code	# Sentences	# Tokens	# Entities	Entities per Sentence	% Entities in Tokens
Amharic	amh	1,750	25,829	3,995	2.3	15.5
Luganda	lug	1,428	33,003	5,039	3.5	15.3
Luo	luo	644	18,577	2,704	4.2	14.6
English	en	14,042	203,621	29,450	2.1	14.5
Nigerian Pidgin	pcm	2,124	52,604	7,392	3.5	14.1
Kinyarwanda	kin	2,116	47,912	6,104	2.9	12.7
Swahili	swa	2,109	56,599	7,161	3.4	12.7
Hausa	hau	1,912	55,010	6,836	3.6	12.4
Igbo	ibo	2,235	42,719	5,294	2.4	12.4
Yorùbá	yor	2,171	56,274	6,324	2.9	11.2
Wolof	wol	1,871	36,805	2,157	1.2	5.9

4 Experiments

Having described our corruption strategies, we now perform our experiments and showcase our results. We consider the three corruption strategies described above, and use four different pre-trained language models, described in Section 4.1. Each run consists of fine-tuning a single pre-trained model on a single language's dataset, either the original one or a corrupted version. We fine-tune models for 25 epochs, as the results were similar to 50 epochs (which Adelani *et al.* [2021] used) and trained much faster. We use a learning rate of $5e-5$, a batch size of 64, and a sequence length of 200. We run all experiments over three seeds and average the results.

We specifically investigate the effect of progressively corrupting data on the performance of each model, measured by the overall F1 score. This simulates the effect of having low data quality (for instance due to incorrect annotations), but allows us to study this in a controlled setting. We do not modify the test datasets at all.

Then, to normalise results across models and languages, we divide each F1 score by the value obtained when training the same model on the full, uncorrupted dataset. This effectively measures what fraction of performance is lost when corrupting data and allows us to transform all of the metrics

Table 2: Information about the different pre-trained language models we use. In the *MasakhaNER Languages* column, we list only the languages the model pre-trained on that are included in the MasakhaNER dataset.

Name	Model Version	Source	Parameters	MasakhaNER Languages
AfriBERTa	`afriberta-large`	Ogueji *et al.* [2021]	126M	amh, hau, ibo, kin, pcm, swa, yor
Afro-XLM-R	`afro-xlmr-base`	Alabi *et al.* [2022]	270M	amh, hau, ibo, kin, pcm, swa, yor
XLM Roberta	`xlm-roberta-base`	Conneau *et al.* [2020]	270M	amh, hau, swa
Multilingual BERT	`bert-base-multilingual-cased`	Devlin *et al.* [2019]	110M	swa, yor

to fall between 0 and 1, resulting in the metrics being comparable across languages and models.

In the *cap sentences* strategy, where we train models on a subset of data, we specifically remove a certain percentage of the data and train the model on the remaining sentences. Since the specific fraction we keep may have an effect, we run this experiment three times, each time with different random selections of data. We average over these permutations and find that the results are very similar across them.

4.1 Different Pre-trained Language Models

We use four different pre-trained language models. We first consider two models developed specifically for low-resourced African languages, AfriBERTa and Afro-XLM-R. The other two models are traditional multilingual models, with the majority of the training datasets consisting of high-resourced languages, XLM-R and mBERT (multilingual BERT). AfriBERTa [Ogueji *et al.*, 2021] was pre-trained on less than 1GB of African language text. Afro-XLM-R [Alabi *et al.*, 2022] used *language adaptive fine-tuning*, where a pre-trained language model is fine-tuned on unlabelled data using the same objective that was used during pre-training. Afro-XLM-R performed this process on 20 languages, 17 of them from Africa, starting from XLM Roberta. XLM Roberta [Conneau *et al.*, 2020] is a high-performing model that was pre-trained on 100 languages. Finally, mBERT [Devlin *et al.*, 2019] used the standard BERT training process on 104 languages, using data from Wikipedia.

We choose the specific model versions to be roughly comparable in terms of the number of parameters. More information about the models is included in Table 2.

4.2 Initial Results

In Table 3, we show the results when each model trains on the entire training dataset. Overall, most models perform well on most languages, with Afro-XLM-R performing the best on average. mBERT, on the other hand, performs the worst overall, with an F1 score of 0 on Amharic, as it was not pre-trained on data containing this script [Adelani *et al.*, 2021].

4.3 How Corruption Affects Performance

Here we compare the three different corruption strategies: (1) deleting a certain fraction of sentences, (2) deleting (i.e. setting to *O*) a certain fraction of labels, and (3) swapping (i.e. replacing with another, but incorrect entity) a certain fraction of labels. The results are shown in Figure 2. The first conclusion we can draw from this is that the number of sentences is far less important than the quality of annotations. In particular, when deleting 90% of the sentences (leaving us with about 10% of the labels of the original dataset), we can still recover around 75% of the performance of training on the entire dataset. When we set 90% of the labels to *O*, however, the performance is much worse, just above 10% of training on the original dataset. Thus, even though the number of labels is roughly equal for each case, having incorrectly labelled data affects the models much more than having fewer sentences that are completely labelled.[1] When we swap labels with incorrect entities, the models perform similarly, but slightly worse compared to replacing these labels with *O*. This suggests that when an annotator is uncertain, it is better to leave an entity out compared to labelling it incorrectly.

Overall, this experiment shows that the fraction of correct annotations is an important factor in NER. This means that, for every amount of labels, having fewer, but completely labelled sentences is significantly better than having the same number of sentences, but with incomplete or incorrect labels.

Table 3: The F1 score when fine-tuning each model on unaltered training data. Bold indicates the best performance per language.

Model Language	AfriBERTa	Afro-XLM-R	XLM-R	mBERT	Average
amh	72.1 (0.9)	**75.9 (1.9)**	71.9 (0.9)	0.0 (0.0)	55.0
en	88.5 (0.3)	**92.8 (0.1)**	92.7 (0.2)	92.6 (0.2)	91.7
hau	90.0 (0.5)	**90.8 (0.4)**	89.8 (0.2)	87.2 (0.5)	89.5
ibo	**87.1 (0.3)**	87.0 (0.6)	83.2 (0.2)	84.7 (0.5)	85.5
kin	74.1 (0.7)	**78.1 (0.2)**	72.5 (1.3)	70.7 (0.5)	73.8
lug	78.7 (0.2)	**81.3 (0.2)**	77.7 (0.4)	79.6 (0.7)	79.3
luo	68.1 (0.9)	69.2 (4.9)	69.4 (2.2)	**71.7 (0.9)**	69.6
pcm	85.5 (0.6)	**89.2 (0.3)**	86.2 (1.5)	88.0 (0.1)	87.2
swa	87.5 (0.6)	**88.3 (0.2)**	87.5 (0.6)	86.0 (0.7)	87.3
wol	61.4 (1.4)	**66.1 (1.6)**	63.9 (0.8)	63.4 (0.9)	63.7
yor	79.3 (0.6)	**80.9 (1.0)**	76.5 (1.1)	78.7 (0.7)	78.8
Average	79.3	81.8	79.2	73.0	78.3

4.4 Performance Across Models and Languages

Having demonstrated that, on average, the quality of data is much more important than the quantity of data, we now investigate how these results differ across pre-trained models and languages. In the top row of Figure 3 we plot the results of each corruption strategy, showing the performance of each model separately. Overall, all models perform similarly, with AfriBERTa and mBERT being slightly more robust to being trained on small amounts of data than the larger XLM-R models.

We study the effect of language on our results in Figure 3

[1] We do verify that when keeping only $X\%$ of the sentences, the fraction of labels we are left with is very close to $X\%$ compared to the original dataset, confirming that labels are mostly distributed uniformly throughout the sentences.

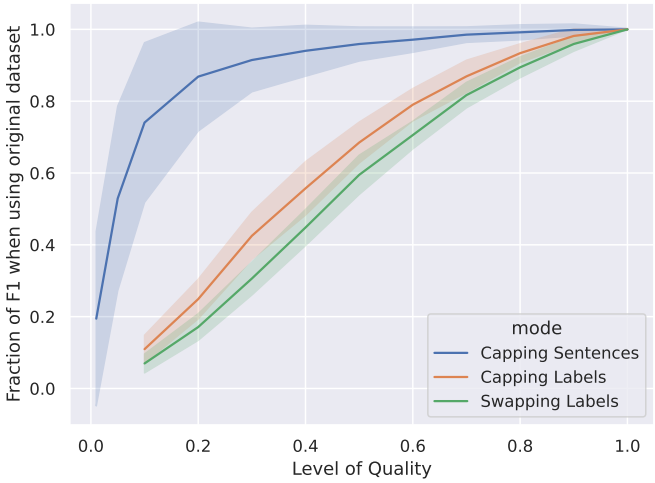


Figure 2: Comparing the result of (blue) deleting sentences, (orange) deleting labels and (green) swapping labels. The X-axis represents the level of quality: 1.0 is the original dataset, whereas 0.1 means that we keep 10% of the sentences or 10% of the labels and corrupt/delete the other 90%. The mean here is shown, with the standard deviation across seeds, languages and models shaded.

(bottom), averaging over the random seeds and pre-trained models. We find that, on average, each language performs similarly. The only exception here is when we reduce the number of sentences the models were trained on, English performs better than average, whereas Luo performs worse. One reason for these observations is that the CONLL dataset is significantly larger than the other languages' datasets, making it perform well when given only a small fraction of the data, as this still corresponds to a large absolute number of sentences. Similarly, Luo has the smallest dataset out of all eleven languages, making it more susceptible to having even less data. This also suggests that having fewer, higher-quality sentences is preferred, but that having too few sentences can drastically reduce performance.

4.5 The Tradeoff Between Quantity and Quality

The results in the previous sections indicate that prioritising data quality and the correctness of annotations is important, more so than the number of sentences we train on. We investigate this further by looking at the relationship between quantity and quality, combining the removal of sentences with the deletion of labels. Since we have shown that our results are mostly consistent across languages and models, we consider only Afro-XLM-R and mBERT as well as three languages, English, Swahili and Luo. We choose these two models due to their differences in number of parameters and pre-training languages. Most languages exhibit roughly similar performance, but English and Luo had slightly different behaviour when deleting sentences, due to their different dataset sizes. Swahili is used as a baseline as it had roughly average behaviour. The results are shown in Figure 4 and they confirm that deleting labels is more detrimental to performance than removing a similar percentage of sentences. For example, for Afro-XLM-R and Swahili, keeping 25% of the sentences but not removing any labels results in close to optimal performance, at 99%. Having 25% of the labels but keeping all of the sentences gives much worse performance, at 37%, even though the overall number of correctly labelled entities is similar.[1] Having 50% of the sentences and 50% of the labels, the results are in-between, just below 70%. However, as can be seen when the fraction of remaining sentences becomes too small (e.g. having 10% or fewer sentences for Luo), the models' performance suffers drastically, regardless of quality. Thus, while having correctly annotated labels is a priority, if we do not have enough data, even perfect-quality annotations will result in poor performance.[2]

4.6 Model Uncertainty

Having considered the effect of corruption on the performance of models, we now look at uncertainty in models' predictions as data corruption increases. Uncertainty has numerous applications related to making decisions regarding model predictions; notably in active learning where the model's uncertainty is used to decide which data points should be labelled [Gal *et al.*, 2017] and self-training where it is used to select training samples from unlabelled data. [Mukherjee and Awadallah, 2020]. We therefore aim to better understand uncertainty specifically when data corruption is taken into consideration. This is because models (when trained on corrupted data) can view the same tokens in similar contexts but with different labels. Due to its simplicity, we use the Monte-Carlo Dropout approach proposed by Gal and Ghahramani [2016] to measure uncertainty. With this approach, we perform multiple forward passes through our model, each predicting slightly different probabilities (over entity classes) due to the stochasticity of the model's dropout layers. We then aggregate the predictions and calculate the resulting entropy [Shannon, 1948].

More formally, we infer an approximation of the predictive distribution of our models $p(y = c|x)$ as:

$$p(y = c|\mathbf{x}) \approx \frac{1}{T}\sum_{t=1}^{T} p\left(y = c|\mathbf{x}, \omega_t\right), \text{ s.t. } \omega_t \sim q(\omega) \quad (1)$$

where $x \in \mathrm{R}^d$ represents the input features of a token and c is an entity class from the available labels. ω is the set of parameters of our model and $q(\omega)$ is a posterior distribution over these parameters. We sample T different sets of parameters from this distribution by performing T different forward passes, using dropout to stochastically zero out some neurons during each forward pass. Finally, we calculate the entropy H of this approximate predictive distribution as follows:

$$H\left[y \mid \mathbf{x}\right] = -\sum_{c} p\left(y = c \mid \mathbf{x}\right) \log p\left(y = c \mid \mathbf{x}\right)$$

For this experiment, we set $T = 50$ in Equation 1 to get a relatively accurate approximation of the predictive distribution while keeping the computational requirements reasonable. We use the unchanged test data to calculate the entropy and consider only the entropy of named entities, as the *O* category had a relatively consistent level of entropy regardless of the corruption.

[2] mBERT shows a similar trend to Afro-XLM-R but, as before, it is slightly more robust to being fine-tuned on small amounts of data.

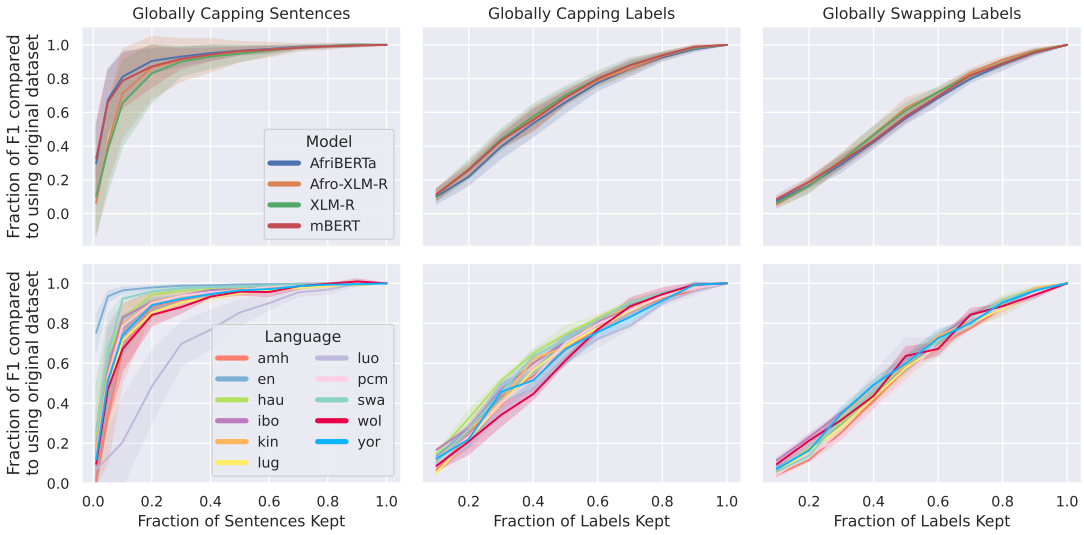


Figure 3: Showcasing the effect of each corruption when isolating each (top) pre-trained model or (bottom) language. We plot the mean and standard deviation over random seeds and the other feature (the language in the top row and the model in the bottom row).

Echoing our previous performance-based results, the entropy does not drastically change as we train on progressively fewer sentences in Figure 5a. Having 20% sentences or fewer does result in a sharp increase in entropy, however. Furthermore, while swapping and capping the labels result in similar performance, the entropy behaviour is significantly different. When we delete most labels, the entropy decreases, as the model is confident that most tokens are not named entities. Conversely, when we swap most of the labels, the entropy is high as, at that point, the labels are effectively random.

These findings suggest that entropy may not be a valid metric to use in cases where our data is heavily corrupted, as it can be high either when not enough data has been seen or when much of the data is randomly labelled. Additionally, we can also find a misleadingly low entropy when much of our data is incorrectly labelled in a systematic way – for example, labelling most entities as *O*.

5 Discussion & Future Work

Our work follows a recent trend of questioning whether more data is always better in NLP, even at the cost of quality [Alabi *et al.*, 2019; Abdulmumin *et al.*, 2022]. In contrast to many of these works, we systematically and quantifiably investigate the effects of reducing the quantity and quality of data. Firstly, our findings demonstrate that the quantity of data used for training does not greatly impact final performance, as we were able to achieve around 80% performance with just 10-20% of the original dataset's sentences. However, removing entity labels, which simulates a reduction in annotation quality, had a significant impact on performance. This supports prior findings that we do not need a large amount of data to perform well when leveraging pre-trained language models [Adelani *et al.*, 2022a]. We do note, however, that when the number of sentences falls below a certain threshold (roughly between 200 and 400 sentences), performance drops significantly, indicating that we do need a minimum amount of data to perform well. Furthermore, while quality has anecdotally been shown to be more important than quantity of data [Alabi *et al.*, 2019; Abdulmumin *et al.*, 2022], here we quantify this effect in NER. Our results imply that when we have the budget to label N entities, using fewer fully-labelled sentences is better than using more sentences that are only partially or incorrectly labelled. Our results also suggest that even modest data-collection and annotation efforts should be able to result in datasets that are large enough to obtain decent performance. Quality, however, is of great importance and should be prioritised in the data creation process.

The second overarching observation we can make is that, in most cases, all models exhibit roughly equal behaviour, in terms of the dropoff in performance, as the level of corruption increases. This ranges from the African language-centric models (AfriBERTa and Afro-XLM-R) to the predominantly high-resourced models (XLM-R and mBERT). This suggests that the behaviour we see here is quite general, as opposed to being specific to just a particular model. The variation across the eleven languages is also remarkably low, again highlighting the consistency of our results. This is notable as we considered languages that were included in the pre-trained models' training data as well as some that were not.

Lastly, we find that replacing entities with *O*, and replacing them with incorrect ones both result in similar performance, with missing labels performing slightly better. However, these two corruptions affect the model's uncertainty in different ways. Progressively removing labels results in the model becoming more certain – predicting most entities as *O*. Swapping labels with incorrect ones, however, results in the model's entropy increasing as the model is much more uncertain. This, coupled with the fact that swapping labels resulted in slightly worse performance, suggests that annotators should rather leave a span of tokens unlabelled if they are uncertain about its label, instead of labelling it incorrectly.

There are numerous avenues for future work. One option

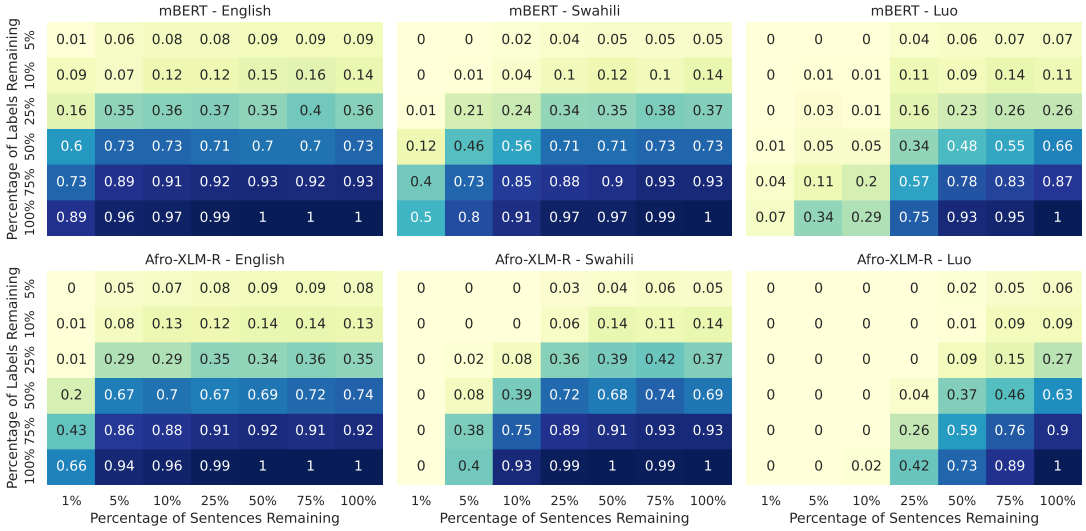
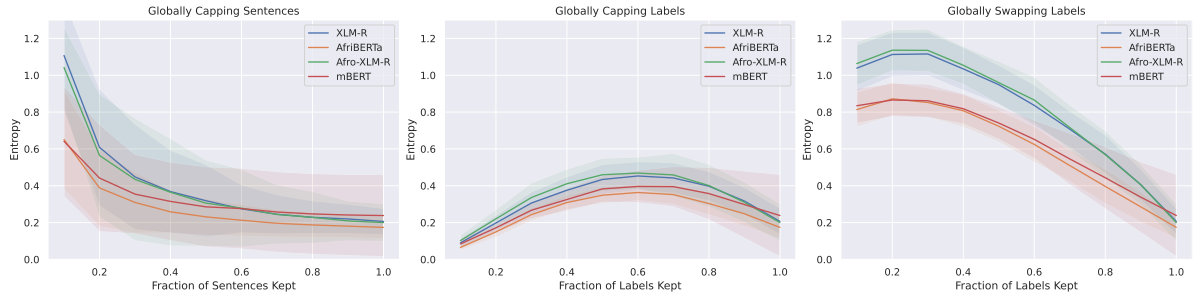


Figure 4: Showing the effect of varying both the number of sentences and the fraction of deleted labels on (top) mBERT and (bottom) Afro-XLM-R for *en, swa* and *luo*. The x-axis shows the percentage of sentences remaining whereas the y-axis lists the percentage of labels remaining. For example, at (50%, 50%), we first remove half of the sentences and then delete half of the labels in the remaining sentences. Each cell contains the fraction of F1 obtained when training on this data compared to training on the original dataset, averaged over 3 seeds.



(a) (b) (c)

Figure 5: Showing the effect of each data corruption strategy on the models' token entropy. We first calculate the mean entropy over tokens for each model, language and seed combination. Here we plot the mean and shade the standard deviation over models, languages and seeds.

would be to expand our work into other NLP tasks such as machine translation or question answering. Developing new corruption strategies that are applicable to other tasks and cover other aspects of quality would also be promising. Investigating the effect of the corruptions on data characteristics, such as the prevalence of rare tokens, would also be valuable. Furthermore, our work motivates research into developing better methods for dealing with corrupted datasets, mitigating some of the negative effects of training on low-quality data. One promising option would be to use active learning, to choose which sentences should be labelled, or verified by a human annotator [Gal *et al.*, 2017].

Finally, we randomly remove sentences and find that this can still result in high performance. Recent work, however, has demonstrated that carefully choosing the data points to train on can result in significantly more data efficiency, requiring vastly less data while achieving comparable performance [Mindermann *et al.*, 2022; Sorscher *et al.*, 2022]. This line of work is promising and has great potential in NLP for low-resourced languages.

6 Conclusion

In this paper, we present a systematic analysis of the impact of data quality and quantity on the performance of pre-trained models in a named entity recognition task for low-resourced languages. By designing multiple corruption strategies and fine-tuning models on datasets with varying degrees of corruption, we are able to provide useful insights into the relationship between data quality and model performance. Our results, which are consistent across pre-trained models and languages, demonstrate that pre-trained models can perform effectively with minimal data and that missing or incorrect annotations have a much greater negative impact than having fewer fully-labelled sentences. The findings of this study have the potential to inform future NER dataset creation efforts and aid NLP practitioners in selecting appropriate datasets for fine-tuning. Ultimately, we believe that this research represents a valuable contribution to the field of NLP and will have a significant impact on the development of NER systems for low-resourced languages.

References

[Abdul-Rauf *et al.* 2012] Sadaf Abdul-Rauf, Mark Fishel, Patrik Lambert, Sandra Noubours, and Rico Sennrich. Extrinsic evaluation of sentence alignment systems. *Workshop on Creating Cross-language Resources for Disconnected Languages and Styles*, 2012.

[Abdulmumin *et al.* 2022] Idris Abdulmumin, Michael Beukman, Jesujoba O. Alabi, Chris Emezue, Everlyn Asiko, Tosin Adewumi, Shamsuddeen Hassan Muhammad, Mofetoluwa Adeyemi, Oreen Yousuf, Sahib Singh, and Tajuddeen Rabiu Gwadabe. Separating grains from the chaff: Using data filtering to improve multilingual translation for low-resourced african languages. In *Proceedings of the Seventh Conference on Machine Translation: Shared Task Papers*, 2022.

[Adelani *et al.* 2021] David Ifeoluwa Adelani, Jade Z. Abbott, Graham Neubig, Daniel D'souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, et al. Masakhaner: Named entity recognition for african languages. *Trans. Assoc. Comput. Linguistics*, 9:1116–1131, 2021.

[Adelani *et al.* 2022a] David Ifeoluwa Adelani, Jesujoba O. Alabi, Angela Fan, Julia Kreutzer, Xiaoyu Shen, Machel Reid, Dana Ruiter, Dietrich Klakow, Peter Nabende, et al. A few thousand translations go a long way! leveraging pre-trained models for african news translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 3053–3070. Association for Computational Linguistics, 2022.

[Adelani *et al.* 2022b] David Ifeoluwa Adelani, Graham Neubig, Sebastian Ruder, Shruti Rijhwani, Michael Beukman, Chester Palen-Michel, Constantine Lignos, Jesujoba O. Alabi, Shamsuddeen Hassan Muhammad, et al. Masakhaner 2.0: Africa-centric transfer learning for named entity recognition. *CoRR*, abs/2210.12391, 2022.

[Alabi *et al.* 2019] Jesujoba O. Alabi, Kwabena Amponsah-Kaakyire, David Ifeoluwa Adelani, and Cristina España-Bonet. Massive vs. curated word embeddings for low-resourced languages. the case of yorùbá and twi. *CoRR*, abs/1912.02481, 2019.

[Alabi *et al.* 2022] Jesujoba O Alabi, David Ifeoluwa Adelani, Marius Mosbach, and Dietrich Klakow. Multilingual language model adaptive fine-tuning: A study on african languages. *CoRR*, abs/2204.06487, 2022.

[Conneau *et al.* 2020] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8440–8451. Association for Computational Linguistics, 2020.

[de Gibert Bonet *et al.* 2022] Ona de Gibert Bonet, Ksenia Kharitonova, Blanca Calvo Figueras, Jordi Armengol-Estapé, and Maite Melero. Quality versus quantity: Building Catalan-English MT resources. In *Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages*, pages 59–69, Marseille, France, June 2022. European Language Resources Association.

[Devlin *et al.* 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019.

[Gal and Ghahramani 2016] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Maria-Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1050–1059. JMLR.org, 2016.

[Gal *et al.* 2017] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *ICML*, volume 70 of *Proceedings of Machine Learning Research*, pages 1183–1192. PMLR, 2017.

[Gascó *et al.* 2012] Guillem Gascó, Martha-Alicia Rocha, Germán Sanchis-Trilles, Jesús Andrés-Ferrer, and Francisco Casacuberta. Does more data always yield better translations? In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 152–161, Avignon, France, April 2012. Association for Computational Linguistics.

[Kreutzer *et al.* 2022] Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, et al. Quality at a glance: An audit of web-crawled multilingual datasets. *Trans. Assoc. Comput. Linguistics*, 10:50–72, 2022.

[Lample and Chaplot 2017] Guillaume Lample and Devendra Singh Chaplot. Playing FPS games with deep reinforcement learning. In Satinder Singh and Shaul Markovitch, editors, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 2140–2146. AAAI Press, 2017.

[Martinus and Abbott 2019] Laura Martinus and Jade Z. Abbott. A focus on neural machine translation for african languages. *CoRR*, abs/1906.05685, 2019.

[Mindermann *et al.* 2022] Sören Mindermann, Jan Markus Brauner, Muhammed Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N. Gomez,

Adrien Morisot, Sebastian Farquhar, and Yarin Gal. Prioritized training on points that are learnable, worth learning, and not yet learnt. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 15630–15649. PMLR, 2022.

[Mukherjee and Awadallah 2020] Subhabrata Mukherjee and Ahmed Awadallah. Uncertainty-aware self-training for few-shot text classification. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 21199–21212. Curran Associates, Inc., 2020.

[Nekoto *et al.* 2020] Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi E. Fasubaa, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddeen Hassan Muhammad, Salomon Kabongo Kabenamualu, Salomey Osei, et al. Participatory research for low-resourced machine translation: A case study in african languages. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 2144–2160. Association for Computational Linguistics, 2020.

[Nekoto *et al.* 2022] Wilhelmina Nekoto, Julia Kreutzer, Jenalea Rajab, Millicent Ochieng, and Jade Abbott. Participatory translations of oshiwambo: Towards sustainable culture preservation with language technology. In *3rd Workshop on African Natural Language Processing*, 2022.

[Ogueji *et al.* 2021] Kelechi Ogueji, Yuxin Zhu, and Jimmy Lin. Small data? no problem! exploring the viability of pretrained multilingual language models for low-resourced languages. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pages 116–126, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[Radford *et al.* 2018] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[Radford *et al.* 2019] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[Sang and Meulder 2003] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of the Seventh Conference on Natural Language Learning, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, Canada, May 31 - June 1, 2003*, pages 142–147. ACL, 2003.

[Shannon 1948] Claude E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27(3):379–423, 1948.

[Sorscher *et al.* 2022] Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari S. Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *CoRR*, abs/2206.14486, 2022.

[Vaswani *et al.* 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.