Evaluating Inclusivity, Equity, and Accessibility of NLP Technology: A Case Study for Indian Languages

Anonymous ACL submission

Abstract

In order for NLP technology to be widely applicable and useful, it needs to be *inclusive* of users across the world's languages, *equitable*, i.e., not unduly biased towards any particular language, and *accessible* to users, particularly in low-resource settings where compute constraints are common. In this paper, we propose an evaluation paradigm that assesses NLP technologies across all three dimensions, hence quantifying the diversity of users they can serve. While inclusion and accessibility have received attention in recent literature, equity is currently unexplored. We propose to address this gap using the Gini coefficient, a well-established metric used for estimating societal wealth inequality. Using our paradigm, we highlight the distressed state of diversity of current technologies for Indian (IN) languages. Our focus on IN is motivated by their linguistic diversity and their large, varied speaker population. To improve upon these metrics, we demonstrate the importance of region-specific choices in model building and dataset creation and also propose a novel approach to optimal resource allocation during fine-tuning. Finally, we discuss steps that must be taken to mitigate these biases and call upon the community to incorporate our evaluation paradigm when building linguistically diverse technologies.

1 Introduction

NLP has seen large advances in recent years driven by the rapid progress in transfer learning (Ruder et al., 2019; Devlin et al., 2019). The benefits of these advances, however, are not equally distributed across the world's languages (Joshi et al., 2020) and users. While linguistic diversity and inclusion have evolved to be a pressing concern today, measures to quantify these are still lacking. The progress of any field is tightly coupled with its evaluation paradigm and the community is incentivized to work on highly visible metrics and benchmarks. In order for users around the world to reap the benefits of NLP technology, we must move from an evaluation that focuses on optimizing raw performance on available test data to a more holistic user-centric evaluation (Ethayarajh and Jurafsky, 2020; Ruder et al., 2021). For multilingual systems, such an evaluation should consider three dimensions: inclusivity, equity, and accessibility.[1]

Inclusivity is important as NLP technology should be available to speakers of any language (European Language Resources Association, 2019). To this end, recent work (Blasi et al., 2021) quantifies inclusivity of NLP technology across the world's languages by weighing task performance for each language based on its speaker population.

Equity is key as we should aim to develop technology that does not discriminate against speakers of any particular language (Kaneko and Bollegala, 2019). State-of-the-art multilingual models in fact have been shown to perform much better in languages with access to many pre-training resources (Hu et al., 2020). To measure such performance inequity across languages, we propose to use the Gini coefficient (Dorfman, 1979), a measure that has been used to represent the income inequality within social groups.

Finally, accessibility is a concern as the fact that NLP technology is performant in a given task and language does not mean that it is usable. State-of-the-art models have been becoming larger and larger (Fedus et al., 2021) and the low-resource setting of many languages often coincide with constraints on computational resources (Ahia et al., 2021). The value a technology provides to a user thus also needs to consider how easily such technology can be run and deployed in practice, which we quantify based on a model's efficiency at runtime, specifically its throughput and memory.

Using our paradigm, we highlight the distressing

[1] We focus on assessing these dimensions on the *language level*. Prior work on equity focuses mainly on subpopulations *within* a language (Katell et al., 2020).

state of diversity in current technologies for Indian (IN) languages. India is a multilingual society with 1369 rationalized languages and dialects being spoken across the country (Chandramouli, 2011). Of these, 22 scheduled languages[2], spoken by almost 97% of the population hold an official recognition and 121 languages have more than 10,000 speakers. Additionally, 21.92% of its population lives below the poverty line (RBI, 2021). Therefore, serving this large varied population justly, requires a multi-faceted effort and basing our case study on IN languages directs the way forward.

We evaluate state-of-the-art models across four standard downstream tasks: *Named Entity Recognition* (NER), *Part-of-Speech Tagging* (POS), *Natural Language Inference* (NLI) and *Question Answering* (QA). We evaluate a range of state-of-the-art models and transfer settings (Hu et al., 2020). We observe that region-specific choices, i.e. region-specific models (Kakwani et al., 2020; Khanuja et al., 2021) and Hindi as source language generally yield the best results. In terms of efficiency, we find that smaller models are preferable for easier, syntactic tasks while larger models have the edge on more complex, semantic tasks.

Our findings, however, also highlight that we are still a long way from building perfectly inclusive and equitable NLP technology. Towards bridging this gap, we explore how we can most effectively fine-tune pre-trained models. Specifically, we propose a fully computational approach to model the space of source and target languages, and derive the optimal allocation of a fixed annotation budget to maximize performance on our proposed metrics.

Our contributions are the following: **1)** We propose a holistic evaluation paradigm that assesses NLP technology based on their inclusivity, equity and accessibility. **2)** Using this paradigm, we evaluate model capabilities for IN languages and quantify their shortcomings. **3)** We propose a novel approach to fine-tune these models with the objective of maximizing performance for the proposed metrics. **4)** We discuss steps that must be taken to mitigate these biases and call upon the community to incorporate our evaluation paradigm when building models to track progress towards building lunguistically inclusive and diverse technologies.

[2]Assamese, Bengali, Bodo, Dogri, Gujarati, Hindi, Kashmiri, Kannada, Konkani, Maithili, Malayalam, Manipuri, Marathi, Nepali, Oriya, Punjabi, Tamil, Telugu, Sanskrit, Santali, Sindhi, Urdu

2 Background and Related Work

Multilingual Models Transformer-based language models (LMs) (Vaswani et al., 2017) trained on massive amounts of text from multiple languages have enabled the inclusion of an unprecedented number of languages in NLP technologies (Conneau et al., 2019; Devlin et al., 2018). However, previous research has shown that these models do not serve all languages equally, with resource-poor languages in the long tail suffering the most (Hu et al., 2020; Lauscher et al., 2020). These models go through a critical step of fine-tuning for the downstream task before being deployed. Several recent works focus on optimal fine-tuning strategies that mitigate transfer gaps and improve overall performance across target languages. Lin et al. (2019) propose a tool that chooses optimal transfer languages based on linguistic features. Lauscher et al. (2020) demonstrate the effectiveness of investing in few-shot in-language training examples. Most recently, Debnath et al. (2021) show that investing in an equal number of fine-tuning instances across target languages performs best. These past approaches however, have all been heuristically designed based on the knowledge and intuition of the experimenter, unlike our proposed method that is purely empirical.

User-centric Evaluation At its core, the need for language diversity in technologies is tied to the people it serves. Previous work (Ethayarajh and Jurafsky, 2020; Ma et al., 2021) has highlighted the need for more transparent and user-centric leaderboard evaluation, reporting practically relevant statistics such as model size, energy efficiency, and inference latency. It is common for speaker populations of under-represented languages to operate in resource-constrained settings. Therefore, in addition to evaluating *linguistic* diversity, we employ efficiency metrics to assess *accessibility* of these technologies. With regards to linguistic diversity, Ruder et al. (2021) highlight the need for more fine-grained evaluation across languages and introduce language-specific leaderboards. Blasi et al. (2021) quantify the value of NLP technology weighed by speaker population and determine utilities of several technologies across the world's languages.

Indian Languages The research community has actively been contributing to the advancement of IN NLP by collecting and open-sourcing data (Kak-

wani et al., 2020; Ramesh et al., 2021; Abraham et al., 2020; Roark et al., 2020; Kunchukuttan et al., 2017; Khanuja et al., 2020a), building region-specific multilingual models (Khanuja et al., 2021; Kakwani et al., 2020; Ramesh et al., 2021) and creating evaluation benchmarks (Kakwani et al., 2020; Khanuja et al., 2020b)[3]. Several of these efforts have been undertaken by AI4Bharat[4], a non-profit open-source community that has additionally been working on developing resources for IN signed languages (Sridhar et al., 2020) and creating input tools to type in IN scripts. Recently, Google Research India launched a question answering (QA) challenge named ChAII[5]. Microsoft Research India has also made significant contributions to IN NLP with several efforts directed towards code-mixed language processing[6] and building tools and datasets for under-represented languages in India[7].

3 Inclusion, Equity and Accessibility

3.1 Inclusion: *Utility, Demand and the Global Metric*

The global metric introduced by Blasi et al. (2021) helps quantify linguistic inclusion. Formally, this metric is composed of the utility of a technology weighed by its demand. The utility u_l of a system for a task and language is its performance normalized by the best possible performance afforded by such a task, i.e.,

$$u_l = \frac{\text{performance}_l}{\text{theoretical max performance}}$$

The best possible performance is dictated by human-level performance achieved for the corresponding task.

Demand d_l is characterized by taking into consideration demographic and linguistic perspectives. Under the demographic perspective, the demand for a given technology in a language is estimated to be proportional to the number of speakers of the language itself n_l ($d_l \propto n_l$). Under the linguistic perspective, the demand across languages is identical ($d_l \propto 1$). These two alternatives, as well as any intermediate combination of them, is parameterized through a single exponent τ:

$$d_l^{(\tau)} = \frac{n_l^\tau}{\sum_{l' \epsilon L} n_{l'}^\tau}$$

where $\tau = 1$ correspond to a demographic notion of demand and $\tau = 0$ to a linguistic one. The global metric can now be defined as:

$$M_\tau = \sum_{l \epsilon L} d_l^{(\tau)} . u_l$$

In essence, $M_\tau = 0$ means that no user benefits from language technology and $M_\tau = 1$ corresponds to each language user enjoying perfect technology. Importantly, the higher the difference in M under the linguistic and demographic notions of utility, the greater is the bias towards languages with large speaker populations.

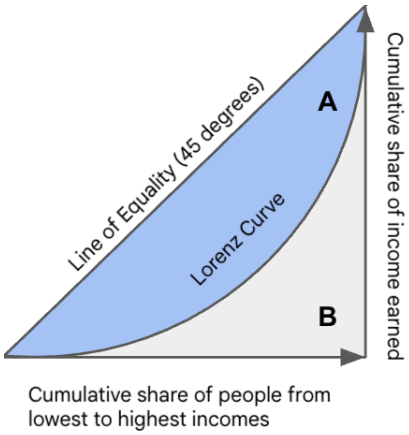

Figure 1: *Graphical Representation* of the Gini coefficient whose value is given by A/(A+B). Please refer to Section 3.2 for more details.

3.2 Equity: *Gini Coefficient*

While linguistic utility assigns equal weight across languages, it does not take into account inequalities in the performance across languages. A model that achieves a performance of 0.9 in Hindi and 0.1 in Tamil is assigned the same linguistic utility as a model that obtains 0.5 in both languages, despite the first one being much less equitable. We propose to use the Gini coefficient to measure such inequity in language performance. The Gini coefficient (Dorfman, 1979) is a measure of statistical dispersion popularly used to quantify income inequality within a social group.

While several past works have highlighted transfer gaps in performance across languages (Hu et al.,

[3] https://github.com/AI4Bharat/indicnlp_catalog maintains a list of resources for Indian NLP.
[4] https://ai4bharat.org/
[5] https://www.kaggle.com/c/chaii-hindi-and-tamil-question-answering
[6] https://www.microsoft.com/en-us/research/project/melange
[7] https://www.microsoft.com/en-us/research/project/ellora

2020), none have quantified this dispersion.[8] The Gini coefficient has several useful properties compared to alternative metrics to compute performance inequity such as the standard deviation, the difference between minimum and maximum, etc: it is a) scale-independent; b) bounded; and c) less influenced by outliers (De Maio, 2007).

The Gini coefficient is mathematically computed based on the Lorenz curve, which plots the relation between population size and the cumulative income earned by that population as shown in Figure 1. To plot the Lorenz curve, individuals are sorted in increasing order of income (x-axis) and their cumulative wealth is plotted on the y-axis. In essence, a point (x, y) indicates that the bottom x% of the population holds y amount of wealth. The line at 45 degrees represents perfect equality of incomes. The Gini coefficient G is then calculated as the ratio of the area that lies between the line of equality and the Lorenz curve (A in Figure 1), over the total area under the line of equality (A + B in Figure 1). If $G = 0$, every person in the population receives an equal percentage of income and if $G = 1$, a single person receives 100% of the income. Since the axes scale from 0 to 1, $A + B = 0.5$. In essence, if the Lorenz curve is represented by the function $Y = L(X)$ then G can be given as:

$$G = \frac{A}{A+B} = 2A = 1 - 2B = 1 - 2\int_0^1 L(X)dX$$

For a population with values y_i, $i = 1 \ldots n$, that are indexed in non-decreasing order ($y_i \leq y_{i+1}$):

$$G = \frac{1}{n}\left(n + 1 - 2\frac{\sum_{i=1}^{n}(n+1-i)y_i}{\sum_{i=1}^{n} y_i}\right)$$

3.3 Accessibility: *Efficiency Score*

Language technology is only beneficial if it can be deployed and accessed by users in a region. We employ efficiency to quantify accessibility as user devices are resource-constrained in many low-resource settings. In line with work on user-centric evaluation (Ethayarajh and Jurafsky, 2020; Ma et al., 2021), we propose to incorporate efficiency into model performance based on two aspects:

Throughput: The throughput of the model is defined as the number of instances it can process per second on a CPU, assuming that GPUs are rarely used for deployment at scale in resource-constrained environments.

[8] Hu et al. (2020) only considered the difference between English and other languages as cross-lingual transfer gap.

Memory Saved: We additionally consider the size of the model as a measure of how expensive a model is to use in practice. Since we wish to minimize this metric, we transform *memory used* into *memory saved* by subtracting it from a maximum available memory of 16 GB (Ma et al., 2021). We show the memory and throughput values for our models in Appendix A.1.

In the efficiency score, we wish to capture the *benefit* associated with per unit increase in *cost*, where *cost* is given by the decrease in throughput or memory saved. The model performance is taken as a proxy for its *benefit*. Let x denote our set of pre-trained models and $M(x)$ denote metric values. Following Ma et al. (2021), we make two key assumptions: **i)** All models lie on the same indifference curve[9]; **ii)** if $M(x_i) > M(x_{i+1})$ and $perf(x_i) > perf(x_{i+1})$, then there exists a model $\langle perf(x_{i+1}), M(x_i) + (M(x_i) - M(x_{i+1}))\rangle$ on the same indifference curve as x_i. Under these assumptions, we can calculate the average benefit-cost ratio (ABCR) for each metric and define $Efficiency(x_i)$ as :

$$Efficiency(x_i) = \sum_M w_M * \frac{M(x_i)}{ABCR(M, perf)}$$

$$ABCR(M, perf) = \overline{BCR}$$

$$BCR = \left\{ \left| \frac{perf(x_i) - perf(x_{i+1})}{M(x_i) - M(x_{i+1})} \right| 1 \leq i < n \right\}$$

where we choose $w_{perf} = 0.5$, $w_{throughput} = 0.25$ and $w_{memory} = 0.25$ as default weights. In practice, these weights can be adjusted as per user requirements based on their constraints to calculate the final efficiency score.

Note that we compute the ABCR value for *each language per task* as opposed to prior work focusing on the task alone. Intuitively, we want to calibrate the efficiency of a technology for each language. For high-resource languages and relatively-simpler tasks, smaller models achieve good performances and scaling up merely leads to a 1–2% performance increase. Here, the ABCR will be low, hence increasing the relative importance of metric M in the efficiency calculation. If we only measure raw performance, larger models are ranked higher, but with efficiency considerations, smaller models

[9] An indifference curve represents the combination of *goods* that confer equal utility to the consumer at all points. In our consideration, these goods are *performance*, *throughput* and *memory saved*, all of which are desirable properties in a model.

Language	as	bn	brx	doi	en	gu	hi	kn	kok	ks	mai	ml
Speakers (in M)	23.6	107.4	1.6	2.8	128.5	60.3	691.6	58.8	2.6	7	14.3	35.6
Language	mni	mr	ne	or	pa	sa	sat	sd	ta	te	ur	-
Speakers (in M)	2.2	99.1	3.4	42.6	36.1	3.1	7.7	3.1	76.6	94.5	63.2	-

Table 1: The number of speakers (in millions) for each of the 22 scheduled languages and English. We take the sum total of first, second and third language speakers for each language.

fare better. As we move to low-resource languages or more complex tasks however, performance differences are significant and using larger models is justified both in terms of utility and efficiency.

3.4 Projected Metrics

An issue when attempting to compute utility for the world's languages (Blasi et al., 2021) or a subset of languages is that data to evaluate models is only available in a few languages. All languages without evaluation data are automatically assigned a score of 0, even though models may obtain passable performance on them due to cross-lingual transfer from related languages. We therefore propose to calculate *projected* estimates of the chosen metrics using our best performing model, which may help inform future data annotation efforts. Assuming that languages from the same family have similar linguistic properties and knowing that end-task performance is largely influenced by the size of pre-training corpora (Lauscher et al., 2020), we calculate these estimates based on two factors: the language family and the availability of unlabeled data. Specifically, for each language without test data, we average the performance of all languages from the same family that are in the same data resource group according to Joshi et al. (2020) and have test data available. If there is no language of the same family in the same resource group, we average the scores of all languages in the same resource group with test data available.[10]

3.5 Optimal Allocation of Annotation Budget

As fine-tuning on a few labeled examples in the target language has shown to improve zero-shot transfer performance, we study how to allocate an annotation budget across a number of source languages S in order to optimize for inclusion and equity across a set of target languages T. Previous work employs a feature-based approach to select a single source language to maximize performance on a target language (Lin et al., 2019) or labels examples across all source languages equally (Debnath et al., 2021). We propose a fully computational approach for modeling the space of source and target languages. This is done by empirically estimating performance of language $\mathrm{t} \in \mathrm{T}$ on a held-out set, when fine-tuned on x labeled instances of language $\mathrm{s} \in \mathrm{S}$, $\forall(\mathrm{s}, \mathrm{t})$ pairs, which follows a power-law distribution (Rosenfeld et al., 2019). We now seek to find the optimal allocation $\{\mathrm{x_s} : \mathrm{s} \in \mathrm{S}\}$ subject to $\sum_{\mathrm{s}\in\mathrm{S}} \mathrm{x_s} \leq \mathrm{X}$ (details in Appendix A.5).

We follow a simple greedy approach to solve this constrained optimization problem as shown in Table 10. Specifically, at each step we allocate our sample to the source language conferring the highest marginal gain to all target languages, which is quantified by the summation of the increase in the global metric and the reduction in Gini.[11] At present, we assign equal weight to each metric but this can be changed according to user preferences.

4 Experiments

4.1 Experimental setup

Languages We base our case-study on the 22 scheduled languages of India spoken by 97% of its population. We also include English, since a sizeable population of 128.5M speakers report English to be their first, second or third language. We show the number of speakers per language in Table 1.

Tasks We select tasks from the XTREME (Hu et al., 2020) benchmark. Dataset details and the human performance (HP) for each task can be found in Table 2. For each task, we only evaluate on IN language test sets.

Models Model selection is motivated by two key factors that we wish to explore in our study: **i)** general v/s region-specific choices; **ii)** model efficiency. We choose IndicBERT, MuRIL and XLM-R, the first two being region-specific models and

[10] An alternative approach is to rely on feature-based performance prediction (Xia et al., 2020; Ye et al., 2021), which we leave for future work.

[11] Future work may consider more complex approaches that consider language relatedness based on work on transfer relationship learning (Zamir et al., 2018; Song et al., 2019).

Task	Dataset	Test Langs.	HP
NER	WikiAnn (Pan et al., 2017; Rahimi et al., 2019)	bn, en, gu, hi, ml, mr, pa, ta, te, ur	97.6
POS	Universal Dependencies v2.6 (Nivre et al., 2018)	en, hi, mr, ta, te, ur	97
NLI	XNLI (Conneau et al., 2018)	en, hi, ur	92.8
QA	XQuAD (Artetxe et al., 2019); TyDiQA-GoldP (Clark et al., 2020)	bn, en, hi, te	91.2; 90.1

Table 2: *Finetuning Tasks and Datasets*. HP denotes the human performance for each task. For QA, HP is 91.2 F1 for XQuAD and 90.1 F1 for TyDiQA.

Language	NER	POS	NLI	QA
English	20,000	21,261	392,702	88,602
Hindi	5,000	13,305	392,702 (*-tran*)	88,602 (*-tran*)

Table 3: *Number of training instances for English and Hindi.* (*-tran*) denotes that the English fine-tuning set has been translated to Hindi.

the third being a state-of-the-art model trained on 100+ languages. We consider both the base and large versions for MuRIL and XLM-R. IndicBERT follows the ALBERT architecture (Lan et al., 2019) and is hence much smaller than the base versions of both models. IndicBERT is trained on 11 IN languages, XLM-R includes 15 and MuRIL is trained on 16 IN languages (details in Appendix A.2).

Fine-tuning Following convention, we initially fine-tune the selected models using training data in English (EN) given the availability of labeled data across tasks. However, several past works have highlighted that this choice is sub-optimal and one can obtain much better performance by transferring from closely related languages (Lauscher et al., 2020; Cotterell and Heigold, 2017; Dong et al., 2015; Turc et al., 2021). To examine its effect in our case-study, we additionally fine-tune models on Hindi (HI) because: **i)** 15 out of 22 languages belong to the same language family as HI (Indo-Aryan); **ii)** we have training data available for all tasks in HI[12]; **iii)** HI has the highest speaker population, which may lead to higher demographic utility and is also a future-safe choice to obtain annotations for any task. Table 3 summarizes training data statistics for EN and HI.

[12]Training sets for NLI and QA have been machine-translated from English, which has been shown to perform similar to human-generated train sets (Turc et al., 2021).

Metric	Train Lang.	Model	NER	PoS	NLI	QA	Average
$M_{\tau=0}\uparrow$ (Linguistic)	English	IndicBERT	16.5	16.1	6.5	5.3	11.1
		$\text{XLM-R}_{\text{base}}$	27.0	21.4	10.3	13.8	18.1
		$\text{MuRIL}_{\text{base}}$	33.4	20.7	10.5	14.9	19.9
		$\text{XLM-R}_{\text{large}}$	28.7	21.9	11.0	15.6	19.3
		$\text{MuRIL}_{\text{large}}$	31.5	21.3	11.1	**15.9**	20.0
	Hindi	IndicBERT	23.7	17.6	6.6	4.8	13.2
		$\text{XLM-R}_{\text{base}}$	30.4	22.4	10.6	13.5	19.2
		$\text{MuRIL}_{\text{base}}$	**34.0**	**22.7**	10.8	14.7	20.6
		$\text{XLM-R}_{\text{large}}$	33.0	22.4	**11.5**	15.2	20.5
		$\text{MuRIL}_{\text{large}}$	33.4	22.4	11.4	15.7	**20.7**
$M_{\tau=1}\uparrow$ (Demographic)	English	IndicBERT	39.2	44.2	36.6	28.4	37.1
		$\text{XLM-R}_{\text{base}}$	59.2	58.1	43.6	49.9	52.7
		$\text{MuRIL}_{\text{base}}$	69.6	54.7	45.5	53.8	55.9
		$\text{XLM-R}_{\text{large}}$	61.2	60.3	46.6	56.6	56.2
		$\text{MuRIL}_{\text{large}}$	68.2	58.6	47.4	**57.9**	58.0
	Hindi	IndicBERT	61.0	61.6	39.8	29.9	48.1
		$\text{XLM-R}_{\text{base}}$	70.3	66.7	45.8	50.6	58.3
		$\text{MuRIL}_{\text{base}}$	**75.1**	**67.3**	46.8	54.7	61.0
		$\text{XLM-R}_{\text{large}}$	74.4	66.8	**49.4**	53.2	60.9
		$\text{MuRIL}_{\text{large}}$	74.8	66.5	49.2	54.6	**61.3**
Gini Coeff. $\downarrow$	English	IndicBERT	0.67	0.81	0.92	0.84	0.81
		$\text{XLM-R}_{\text{base}}$	0.61	0.76	0.88	0.83	0.77
		$\text{MuRIL}_{\text{base}}$	0.59	0.76	0.88	0.83	**0.76**
		$\text{XLM-R}_{\text{large}}$	0.6	0.75	0.88	0.83	0.77
		$\text{MuRIL}_{\text{large}}$	0.59	0.76	0.88	0.83	0.77
	Hindi	IndicBERT	0.68	0.8	0.91	0.83	0.81
		$\text{XLM-R}_{\text{base}}$	0.59	0.75	0.87	0.83	**0.76**
		$\text{MuRIL}_{\text{base}}$	0.59	0.75	0.87	0.83	**0.76**
		$\text{XLM-R}_{\text{large}}$	0.59	0.76	0.88	0.83	0.77
		$\text{MuRIL}_{\text{large}}$	0.59	0.75	0.87	0.83	**0.76**
Efficiency $\uparrow$	English	IndicBERT	22.3	37.4	37.3	27.1	31.0
		$\text{XLM-R}_{\text{base}}$	32.4	41.2	39.1	37.2	37.4
		$\text{MuRIL}_{\text{base}}$	39.5	40.0	39.1	37.2	39.8
		$\text{XLM-R}_{\text{large}}$	33.1	41.6	40.4	41.2	39.1
		$\text{MuRIL}_{\text{large}}$	36.4	40.6	40.9	**42.0**	39.9
	Hindi	IndicBERT	31.3	40.8	37.6	25.6	33.8
		$\text{XLM-R}_{\text{base}}$	36.2	43.1	40.1	36.4	39.0
		$\text{MuRIL}_{\text{base}}$	**40.2**	**43.7**	41.0	39.4	**41.1**
		$\text{XLM-R}_{\text{large}}$	37.9	42.6	42.3	40.0	40.7
		$\text{MuRIL}_{\text{large}}$	38.5	42.5	**41.9**	41.3	**41.1**

Table 4: *Zero-shot fine-tuning results.* Overall, $\text{MuRIL}_{\text{large}}$ scores highest on the utility metrics, the Gini coefficient is relatively high across all models and both $\text{MuRIL}_{\text{base}}$ and $\text{MuRIL}_{\text{large}}$ are, on average, equal with regards to efficiency. Note that the metrics are computed considering all 23 languages as detailed in Section 4.1. More discussions in Section 4.2.

4.2 Zero-shot transfer results

Where are we today? We report results of fine-tuning models on EN and HI in Table 4. Overall, the linguistic and demographic global utility metric is highest for $\text{MuRIL}_{\text{large}}$, when fine-tuned on HI. Generally, the linguistic metric is much worse than the demographic one, indicating that past efforts have been skewed towards populous languages, leaving under-represented languages behind. We also observe that utility increases with region-specific choices, both in pre-training and fine-tuning. The Gini coefficient remains relatively high at around 0.76 even for the best models, which highlights the disparity in performance even among languages within a single region[13]. For comparison, for OECD countries from 2008–2009, the Gini coefficient on income for the entire population ranged between 0.34 and

[13]For completeness, we calculate the Gini coefficient only across languages with evaluation data in Appendix A.3.

Metric	NER	PoS	NLI	QA	Average
$M_{\tau=0} \uparrow$	51.9	60.5	18.5	50.7	45.4
$M_{\tau=1} \uparrow$	81.0	81.9	58.8	79.1	75.2
Gini Coeff. $\downarrow$	0.31	0.29	0.79	0.4	0.44

Table 5: *Projected Metrics* that include estimated $\text{MuRIL}_{\text{large}}$ performances for all languages supported by the model. Refer to Section 4.2 for more details.

0.53 while the Gini coefficient for the entire world has been estimated to be between 0.61 and 0.68 (Hillebrand et al., 2009; Klugman and Nations, 2010).

How to handle the lack of evaluation data? In large part, this can be attributed to the absence of evaluation sets across tasks, with as few as 3 (out of 23) languages having test sets for XNLI. As detailed in Section 3.4, languages with no test data are assigned a score of zero, even if models would obtain non-zero performance. Hence, we calculate projected estimates using our best-performing model $\text{MuRIL}_{\text{large}}$ in Table 5. While there is a significant increase across all metrics, the absolute values are still low with the linguistic utility being below 50%.

How accessible are these models? With regards to the efficiency metric, averaging across languages and tasks, $\text{MuRIL}_{\text{large}}$ and $\text{MuRIL}_{\text{base}}$ perform equally. $\text{MuRIL}_{\text{base}}$ has a higher efficiency score (*1-3 points*) for simpler token-level tasks like NER and POS, while $\text{MuRIL}_{\text{large}}$ has higher scores on complex, semantic tasks like NLI (*<1 point*) and QA (*2-5 points*). Larger, more expressive models may thus be preferable in the latter cases despite being costlier on the accessibility front, since smaller models cannot obtain good performance. We illustrate per language efficiency scores in Appendix A.1 and discuss fine-grained observations.

What is the way forward? Overall, the absolute values of the global metric and the Gini coefficient indicate that there lies great potential in both increasing the utility of our models and making them more equitable. Since model performances largely reflect the amount of raw data used in pre-training (Lauscher et al., 2020), creating equitable unlabeled data resources would alleviate these issues. However, this is an ambitious undertaking that is extremely resource intensive and can certainly not be achieved for 6500 languages in the near future. We thus investigate how limited amounts of data can be used to maximally improve utility and equity during fine-tuning.

Metric	Budget	Model	Fine-tuning Strategy			
			English	Hindi	Egalitarian	Greedy
$M_{\tau=0} \uparrow$	1,000	$\text{XLM-R}_{\text{large}}$	24.8	28.8	31.1	31.8
		$\text{MuRIL}_{\text{large}}$	27.5	31.6	**35.1**	**35.1**
	5,000	$\text{XLM-R}_{\text{large}}$	27.3	33.0	35.8	35.3
		$\text{MuRIL}_{\text{large}}$	30.4	33.4	37.2	**37.5**
	10,000	$\text{XLM-R}_{\text{large}}$	26.8	-	36.4	36.6
		$\text{MuRIL}_{\text{large}}$	33.8	-	38.1	**38.5**
$M_{\tau=1} \uparrow$	1,000	$\text{XLM-R}_{\text{large}}$	54.0	66.2	65.4	65.3
		$\text{MuRIL}_{\text{large}}$	60.4	71.3	**74.1**	73.6
	5,000	$\text{XLM-R}_{\text{large}}$	59.4	74.4	75.4	75.7
		$\text{MuRIL}_{\text{large}}$	65.4	74.8	78.2	**78.3**
	10,000	$\text{XLM-R}_{\text{large}}$	59.0	-	77.6	77.6
		$\text{MuRIL}_{\text{large}}$	70.5	-	79.6	**79.9**
Gini Coeff. $\downarrow$	1,000	$\text{XLM-R}_{\text{large}}$	0.6	0.6	0.59	0.59
		$\text{MuRIL}_{\text{large}}$	0.6	0.6	**0.58**	**0.58**
	5,000	$\text{XLM-R}_{\text{large}}$	0.6	0.59	0.59	0.59
		$\text{MuRIL}_{\text{large}}$	0.59	0.59	**0.58**	**0.58**
	10,000	$\text{XLM-R}_{\text{large}}$	0.61	-	0.59	0.59
		$\text{MuRIL}_{\text{large}}$	0.59	-	**0.58**	**0.58**
Gini Coeff. $\downarrow$ (10 languages)	1,000	$\text{XLM-R}_{\text{large}}$	0.087	0.07	0.061	0.06
		$\text{MuRIL}_{\text{large}}$	0.079	0.071	0.044	**0.039**
	5,000	$\text{XLM-R}_{\text{large}}$	0.088	0.057	0.049	0.058
		$\text{MuRIL}_{\text{large}}$	0.062	0.066	0.042	**0.039**
	10,000	$\text{XLM-R}_{\text{large}}$	0.011	-	0.051	0.052
		$\text{MuRIL}_{\text{large}}$	0.05	-	0.04	**0.033**

Table 6: *Performance on NER under different annotation budgets*. We observe that the greedy approach (Section 3.5) performs best across all metrics. We also report the Gini Coeff. calculated across 10 languages for which we have test data. Note that the HI train set has 5,000 examples only. Details in Section 4.3.

4.3 Few-shot results

Problem Formulation For few-shot fine-tuning, we focus on NER where sufficient labeled training data for seven IN languages is available. We employ the source languages $S = \{bn, en, hi, ml, mr, ta, ur\}$ and seek to optimize metrics on the target languages $T = \{bn, en, gu, hi, ml, mr, pa, ta, te, ur\}$. In each setting, we have a limited annotation budget, which we can divide among the source languages. We compare against several competitive baselines: **i)** using only examples from EN or HI respectively; **ii)** distributing the annotation budget in an egalitarian (uniform) way across all source languages (Debnath et al., 2021); **iii)** the greedy approach proposed in Section 3.5. For the greedy approach, we illustrate the best-fit curves for each (s, t) pair in Appendix A.5 (Table 11). As the original calculation of the Gini coefficient takes into account all 23 IN languages, we also calculate the metric over our 10 target languages only, to observe how it differs across baselines.[14]

[14] In the original calculation, languages with no test sets have zero performance. Therefore, dispersion in the performances of our target language set is not as effectively captured and differences across alternative approaches are not observable.

Results We show the results under various annotation budgets in Table 6. Overall, we find that our method yields a higher global metric under most budgets (5 of 6 cases) and also yields a lower Gini coefficient under *all* budget schemes. The optimal allocations for each budget are shown in Table 12. As we can see, the greedy algorithm converges to a solution that is close to uniform. This provides further evidence for the benefits of an egalitarian distribution of annotation budget in order to maximize performance across all languages as the expected marginal gain for languages that have been underrepresented during training will be highest. Both the egalitarian and greedy approaches significantly outperform fine-tuning on EN or HI only, where the former approaches outperform fine-tuning on 10,000 examples of EN with a limited budget of 1,000 examples by 1–3%.

5 Discussion

Building evaluation datasets Having uncovered the linguistic inequity and exclusivity of current NLP technologies, we seek to identify practical measures we can take in order to mitigate these biases. As a first step, it is paramount to build representative evaluation sets for all languages as they are required to accurately measure utility and equity. Out of the 23 languages in our case study, most do not have evaluation data across tasks despite holding official recognition and being spoken by 97% of the population. In light of the benefits of an egalitarian data distribution during few-shot learning, we also recommend the collection of small amounts of data across many languages for training, in order to maximize marginal gain.These datasets should be collected at the grass-roots level, involving the community they need to serve to capture culturally relevant phenomenon. A prime example of this is the Masakhane organisation[15] steering efforts towards data collection in African languages, involving the local community. Incentivizing rural, low-income workers to provide for such data also serves as a viable source of supplementary income, and does not degrade dataset quality (Abraham et al., 2020).

Trading off multilinguality and regionality From a modeling perspective, multilingual pre-trained models have been instrumental to NLP systems supporting an unprecedented number of languages, because of their zero-shot transfer capabilities. However, while these are a big step towards linguistic inclusion, they are subject to limitations such as highly skewed pre-training distributions and limited transfer to under-represented languages (Hu et al., 2020; Lauscher et al., 2020), a bias towards the source language, and sub-optimal tokenization (Wang et al., 2021). A way to combat these issues is to make region-specific choices, both in pre-training and fine-tuning, as observed in Section 4.2. Localizing the problem also enables one to incorporate linguistic expertise and provide support for culturally relevant phenomena like transliteration or code-mixing. Despite this, we must be wary of excessive fragmentation in pre-training as it leads to higher maintenance costs and there is a possibility that these benefits will be overcome with advances in compute and model capacity in the near-future. Optimal fine-tuning however, is promising, as evidenced in Section 4.3 where we observe significant gains in moving away from the zero-shot paradigm. There is still a lot of room for improvement, however, as the best linguistic utility is still less than 40%.

6 Conclusion

We have proposed a framework for the evaluation of NLP technology based on inclusivity, equity, and accessibility. To quantify equity, we have proposed to use the Gini coefficient, a standard metric to measure income inequality within social groups. Focusing on Indian languages, we have assessed to what extent several modeling and data choices affect the value NLP technology confers to users, highlighting the importance of region-specific choices and efficient models. We have also proposed an algorithmic method for resource allocation for task-specific fine-tuning, which outperforms a purely egalitarian distribution of data labeling. Finally, we highlight the importance of building representative evaluation sets from the grass-roots level to enable tracking progress, and discuss how even with the best modeling strategies, we have a long road ahead in building inclusive, equitable systems. While region-specific choices help to a certain extent, building a single global multilingual model without compromising on the three metrics is something we should move towards in the future. We sincerely hope our evaluation paradigm aids in tracking the community's progress in building linguistically diverse technologies.

[15] https://www.masakhane.io/

References

Basil Abraham, Danish Goel, Divya Siddarth, Kalika Bali, Manu Chopra, Monojit Choudhury, Pratik Joshi, Preethi Jyoti, Sunayana Sitaram, and Vivek Seshadri. 2020. Crowdsourcing speech data for low-resource languages from low-income workers. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 2819–2826.

Orevaoghene Ahia, Julia Kreutzer, and Sara Hooker. 2021. The Low-Resource Double Bind: An Empirical Study of Pruning for Low-Resource Machine Translation. In *Findings of EMNLP 2021*.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2019. On the cross-lingual transferability of monolingual representations. *arXiv preprint arXiv:1910.11856*.

Damián Blasi, Antonios Anastasopoulos, and Graham Neubig. 2021. Systematic inequalities in language technology performance across the world's languages. *arXiv preprint arXiv:2110.06733*.

Chandramouli. 2011. Census of india 2011 provisional population totals. *New Delhi: Office of the Registrar General and Census Commissioner*.

Jonathan H Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. Tydi qa: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. *arXiv preprint arXiv:1809.05053*.

Ryan Cotterell and Georg Heigold. 2017. Cross-lingual, character-level neural morphological tagging. *arXiv preprint arXiv:1708.09157*.

Fernando G De Maio. 2007. Income inequality measures. *Journal of Epidemiology & Community Health*, 61(10):849–852.

Arnab Debnath, Navid Rajabi, Fardina Fathmiul Alam, and Antonios Anastasopoulos. 2021. Towards more equitable question answering systems: How much more data do you need? *arXiv preprint arXiv:2105.14115*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL 2019*.

Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1723–1732.

Robert Dorfman. 1979. A formula for the gini coefficient. *The review of economics and statistics*, pages 146–149.

Kawin Ethayarajh and Dan Jurafsky. 2020. Utility is in the Eye of the User: A Critique of NLP Leaderboards. In *Proceedings of EMNLP 2020*.

European Language Resources Association. 2019. BLT4All: Language Technologies for All. `https://lt4all.elra.info/en/`. [Online; accessed Dec. 2019.].

William Fedus, Barret Zoph, and Noam Shazeer. 2021. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *arXiv preprint arXiv:2101.03961*.

Evan Hillebrand et al. 2009. Poverty, growth and inequality over the next 50 years. In *Expert Meeting on How to feed the World in*, volume 2050, pages 2012–02.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A Massively Multilingual Multi-task Benchmark for Evaluating Cross-lingual Generalization. In *Proceedings of ICML 2020*.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The State and Fate of Linguistic Diversity and Inclusion in the NLP World. In *Proceedings of ACL 2020*.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, NC Gokul, Avik Bhattacharyya, Mitesh M Khapra, and Pratyush Kumar. 2020. inlpsuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for indian languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 4948–4961.

Masahiro Kaneko and Danushka Bollegala. 2019. Gender-preserving debiasing for pre-trained word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1641–1650, Florence, Italy. Association for Computational Linguistics.

Michael Katell, Meg Young, Dharma Dailey, Bernease Herman, Vivian Guetler, Aaron Tam, Corinne Binz,

Daniella Raz, and P. M. Krafft. 2020. Toward situated interventions for algorithmic equity: Lessons from the field. *FAT* 2020 - Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 45–55.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, et al. 2021. Muril: Multilingual representations for indian languages. *arXiv preprint arXiv:2103.10730*.

Simran Khanuja, Sandipan Dandapat, Sunayana Sitaram, and Monojit Choudhury. 2020a. A new dataset for natural language inference from code-mixed conversations. *arXiv preprint arXiv:2004.05051*.

Simran Khanuja, Sandipan Dandapat, Anirudh Srinivasan, Sunayana Sitaram, and Monojit Choudhury. 2020b. Gluecos: An evaluation benchmark for code-switched nlp. *arXiv preprint arXiv:2004.12376*.

Jeni Klugman and Development Programme United Nations. 2010. *The real wealth of nations: pathways to human development*. Palgrave Macmillan.

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2017. The iit bombay english-hindi parallel corpus. *arXiv preprint arXiv:1710.02855*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot cross-lingual transfer with multilingual transformers. *arXiv preprint arXiv:2005.00633*.

Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. 2019. Choosing Transfer Languages for Cross-Lingual Learning. In *Proceedings of ACL 2019*.

Zhiyi Ma, Kawin Ethayarajh, Tristan Thrush, Somya Jain, Ledell Wu, Robin Jia, Christopher Potts, Adina Williams, and Douwe Kiela. 2021. Dynaboard: An evaluation-as-a-service platform for holistic next-generation benchmarking. *arXiv preprint arXiv:2106.06052*.

Joakim Nivre, Mitchell Abrams, Željko Agić, Lars Ahrenberg, and Antonsen et al. 2018. Universal Dependencies 2.2. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for ner. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Divyanshu Kakwani, Navneet Kumar, et al. 2021. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. *arXiv preprint arXiv:2104.05596*.

Reserve Bank of India RBI. 2021. Handbook of statistics on indian economy.

Brian Roark, Lawrence Wolf-Sonkin, Christo Kirov, Sabrina J. Mielke, Cibu Johny, Işin Demirşahin, and Keith Hall. 2020. Processing South Asian languages written in the Latin script: the Dakshina dataset. In *Proceedings of The 12th Language Resources and Evaluation Conference (LREC)*, pages 2413–2423.

Jonathan S Rosenfeld, Amir Rosenfeld, Yonatan Belinkov, and Nir Shavit. 2019. A constructive prediction of the generalization error across scales. *arXiv preprint arXiv:1909.12673*.

Sebastian Ruder, Noah Constant, Jan Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Graham Neubig, and Melvin Johnson. 2021. XTREME-R: Towards More Challenging and Nuanced Multilingual Evaluation. In *Proceedings of EMNLP 2021*.

Sebastian Ruder, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer learning in natural language processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18, Minneapolis, Minnesota. Association for Computational Linguistics.

Jie Song, Yixin Chen, Xinchao Wang, Chengchao Shen, and Mingli Song. 2019. Deep model transferability from attribution maps. In *Advances in Neural Information Processing Systems*.

Advaith Sridhar, Rohith Gandhi Ganesan, Pratyush Kumar, and Mitesh Khapra. 2020. Include: A large scale dataset for indian sign language recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1366–1375.

Iulia Turc, Kenton Lee, Jacob Eisenstein, Ming-Wei Chang, and Kristina Toutanova. 2021. Revisiting the primacy of english in zero-shot cross-lingual transfer. *arXiv preprint arXiv:2106.16171*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Xinyi Wang, Sebastian Ruder, and Graham Neubig. 2021. Multi-view Subword Regularization. In *Proceedings of NAACL 2021*.

Mengzhou Xia, Antonios Anastasopoulos, Ruochen Xu, Yiming Yang, and Graham Neubig. 2020. Predicting performance for natural language processing tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8625–8646, Online. Association for Computational Linguistics.

Zihuiwen Ye, Pengfei Liu, Jinlan Fu, and Graham Neubig. 2021. Towards more fine-grained and reliable NLP performance prediction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3703–3714, Online. Association for Computational Linguistics.

Amir R Zamir, Alexander Sax, William Shen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. 2018. Taskonomy: Disentangling Task Transfer Learning. In *Proceedings of CVPR 2018*.

A Appendix

A.1 Efficiency

As detailed in Section 3.3, we report the throughput and memory for each model and task in Table 7. For NLI, POS and NER, the maximum sequence length is 128 and for QA it's 384.

We plot per-language efficiency scores for two tasks namely POS and QA in Figure 2 for all models when fine-tuned on English. For most languages, efficiency scores drop when we move from the base to large versions in POS, and for EN even the smallest model, IndicBERT, has an efficiency score similar to large models. However for QA, we observe a uniform gain in efficiency across languages as we move to larger models.

A.2 Pre-training Languages

In Section 4.1, we choose IndicBERT, MuRIL and XLM-R as pre-trained multilingual models to base our analysis upon. IndicBERT is trained on 11 IN languages that include Assamese (as), Bengali (bn), Gujarati (gu), Hindi (hi), Kannada (kn), Malayalam (ml), Marathi (mr), Oriya (or), Punjabi (pa), Tamil (ta), Telugu (te). XLM-R includes 15 IN languages in training with the addition of Nepali (ne), Sanskrit (sa), Sindhi (sd) and Urdu (ur) over IndicBERT and MuRIL is trained on 16 IN languages, with the addition of Kashmiri (ks) over XLM-R.

Model	Metric	NER	PoS	NLI	QA
IndicBERT	Memory	129MB			
	Throughput	22.61	20.18	22.91	10.48
	Perf (HI)	59.1	78.5	70.3	49.5
$\text{XLM-R}_{\text{base}}$	Memory	1040MB			
	Throughput	24.39	23.15	26.37	14.91
	Perf (HI)	68.4	82.3	78.5	70.7
$\text{MuRIL}_{\text{base}}$	Memory	909MB			
	Throughput	23.81	23.06	26.23	15.65
	Perf (HI)	75.7	83.4	79.7	76.6
$\text{XLM-R}_{\text{large}}$	Memory	2090MB			
	Throughput	9.35	9.95	10.35	4.1
	Perf (HI)	74.0	81.8	84.5	79.0
$\text{MuRIL}_{\text{large}}$	Memory	1890MB			
	Throughput	9.8	9.93	10.5	4.22
	Perf (HI)	76.4	81.5	84.1	81.7

Table 7: The throughput is given by the number of instances processed per second by the fine-tuned models on CPU.

Metric	Train Lang.	Model	NER	PoS	NLI	QA	Average
Gini Coeff. ↓	English	IndicBERT	0.155	0.107	0.051	0.091	0.101
		$\text{XLM-R}_{\text{base}}$	0.095	0.067	0.058	0.048	0.067
		$\text{MuRIL}_{\text{base}}$	0.047	0.086	0.048	0.03	**0.052**
		$\text{XLM-R}_{\text{large}}$	0.084	0.06	0.049	0.026	0.055
		$\text{MuRIL}_{\text{large}}$	0.051	0.086	0.051	0.027	0.057
	Hindi	IndicBERT	0.173	0.073	0.004	0.041	0.073
		$\text{XLM-R}_{\text{base}}$	0.067	0.037	0.039	0.046	0.047
		$\text{MuRIL}_{\text{base}}$	0.062	0.032	0.036	0.012	**0.035**
		$\text{XLM-R}_{\text{large}}$	0.057	0.04	0.033	0.029	0.04
		$\text{MuRIL}_{\text{large}}$	0.065	0.057	0.033	0.014	0.042

Table 8: *Gini Coefficient* for all models calculated only across languages having evaluation sets for each task.

A.3 Gini Coefficient

As mentioned in Section 4.2, calculating the gini coefficient across all 23 languages doesn't reflect the dispersion in performances across languages for which we have test sets. To compare between baselines, we additionally report the Gini coefficient evaluated only across those languages for which we have test sets as shown in Table 8. We observe that region-specific choices ($\text{MuRIL}_{\text{base}}$ fine-tuned on HI) lead to the lowest value, similar to what we observe with the global metric.

A.4 Fine-tuning Details

We fine-tune all models using the hyperparameters mentioned in Table 9 for each task and model consistently throughout the paper. We make an exception for IndicBERT when fine-tuning on NER, where we fine-tune for 15 epochs instead of 10, to reach convergence.

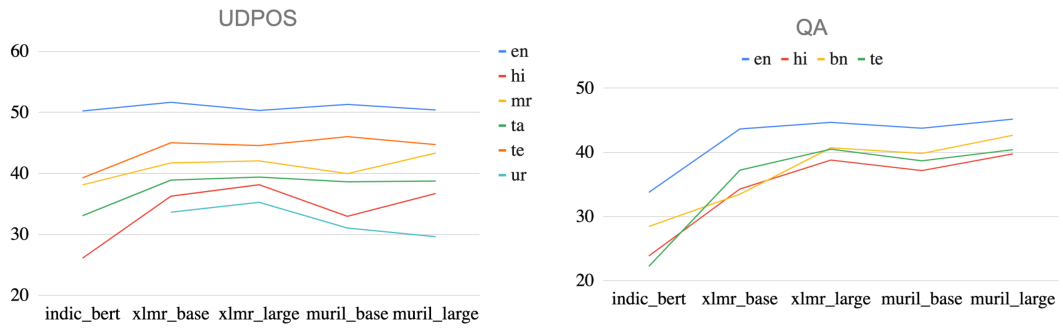

Figure 2: Efficiency scores per-language, per-task when models are fine-tuned on EN

Task	Batch Size	Learning Rate	No. of Epochs	Warmup Ratio	Max. seq. Length
NER	32	2e-5	10	0.1	128
POS	32	2e-5	10	0.1	128
NLI	64	2e-5	3	0.1	128
QA	32	3e-5	2	0.1	384

Table 9: Hyperparameter details for each fine-tuning task

A.5 Budget Allocation

In Section 3.5, we describe an empirical budget allocation scheme for fine-tuning of pre-trained models that can jointly optimize on our proposed metrics. We follow a greedy approach to solve this problem, as shown in Table 10. In this paper, we solve this for one task, namely NER, but the methodology proposed is generally extensible to any task and combination of languages since it is purely empirical. We select seven source languages for which we have enough training data and fine-tune $\mathrm{MuRIL}_{\mathrm{large}}$ and $\mathrm{XLM\text{-}R}_{\mathrm{large}}$ for each of these source languages independently, for two epochs. During fine-tuning, we evaluate on each of our target languages after every 10 steps of training. Given our batch-size is 32, we gather data-points at a step size of 320 training instances. Consequently, say we have 5000 training instances for a source language, we gather approximately 30 sample points for that source language and any target language. Using these, we plot best-fit curves for $\forall(\mathrm{s},\mathrm{t})$ pairs using the *scipy.optimize.curve_fit* package. Given a function, $\mathrm{f}(\mathrm{x})$, *curve_fit* uses non-linear least squares to fit $\mathrm{f}(\mathrm{x})$ to the observed data-points. We define $\mathrm{f}(\mathrm{x})_{\mathrm{s,t}} = \mathrm{a}_{\mathrm{s,t}} + \mathrm{b}_{\mathrm{s,t}} * \mathrm{x}^{-\mathrm{c}_{\mathrm{s,t}}}$, because the relation between model performance and training data follows a power-law distribution (Rosenfeld et al., 2019). The best-fit curves for each source and target pair are shown in Table 11. The visualizations of the best-fit curves for a sample training language (Tamil) are shown in Figures 3, 4. Having determined constant values $\{\mathrm{a}_{\mathrm{s,t}}, \mathrm{b}_{\mathrm{s,t}}, \mathrm{c}_{\mathrm{s,t}}\}\ \forall(\mathrm{s},\mathrm{t})$ independently, we proceed with finding the optimal allocation using the algorithm described in Table 10. We solve this for three different budgets, i.e., 1,000; 5,000 and 10,000 and the optimal allocations for each budget are shown in Table 12.

Greedy Algorithm

Input: Fine-tuning labeled data $\forall s \epsilon S$. A fixed budget of labeled data instances X
Initialize: Set the total number of allocated instances to zero, i.e., $\text{allocated} = 0$, the number of allocated samples for each source language to zero, i.e. $\text{samples}[s] = 0 \forall s \epsilon S$, the current global metric for each source language to -inf, i.e. $\text{current\_gm}[s] = -\text{inf} \forall s \epsilon S$ and the current gini coefficient for each source language to 1, i.e. $\text{current\_gini}[s] = 1 \forall s \epsilon S$
while $\text{allocated} < X$ **do**
  $\text{highest\_marginal\_gain} = 0$
  for s in S **do**
    $\text{gm}_s = \sum_{t \epsilon T} d_t^{(\tau)} * (a_{s,t} + b_{s,t} * (\text{samples}[s] + 1)^{-c_{s,t}})$
    $\text{gini}_s = F[\text{abs}(\text{performance}_{s,t}(\text{samples}[s] + 1)) \forall t \epsilon T]$
    $\Delta\text{gm}_s = \text{gm}_s - \text{current\_gm}[s]$
    $\Delta\text{gini}_s = \text{current\_gini}[s] - \text{gini}_s$
    $\text{marginal\_gain} = \alpha * \Delta\text{gm}_s + \beta * \Delta\text{gini}_s$
    if $\text{marginal\_gain} > \text{highest\_marginal\_gain}$ **do**
      $\text{highest\_marginal\_gain} = \text{marginal\_gain}$
      $\text{best\_language} = s$
      $\text{best\_gm} = \text{gm}_s$
      $\text{best\_gini} = \text{gini}_s$
    end if
  end for
  $\text{samples}[s] = \text{samples}[s] + 1$
  $\text{allocated} = \text{allocated} + 1$
  $\text{current\_gm}[\text{best\_language}] = \text{best\_gm}$
  $\text{current\_gini}[\text{best\_language}] = \text{best\_gini}$
end while

Table 10: A greedy approach to solve constrained optimization for the budget allocation problem as described in Appendix A.5

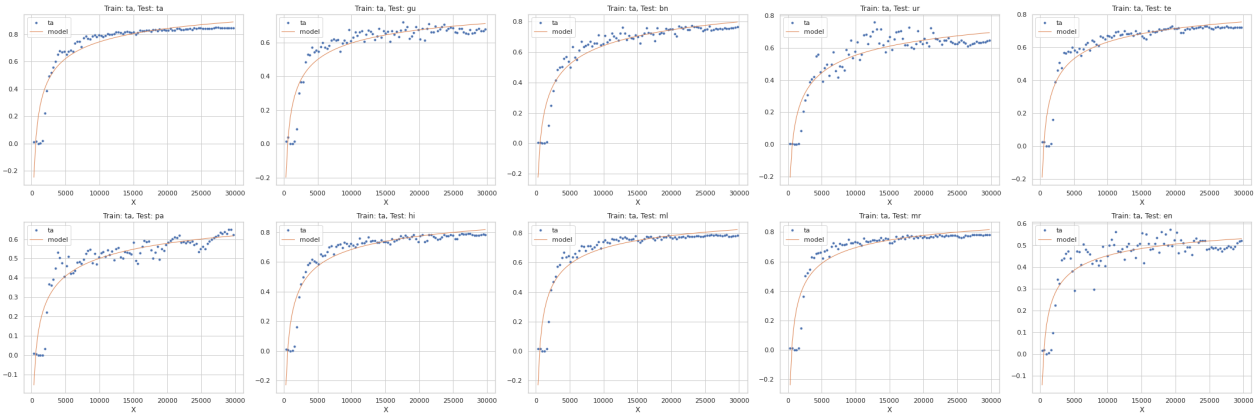

Figure 3: *Best-fit curves for XLM-R* when fine-tuned on Tamil for each of the target languages.

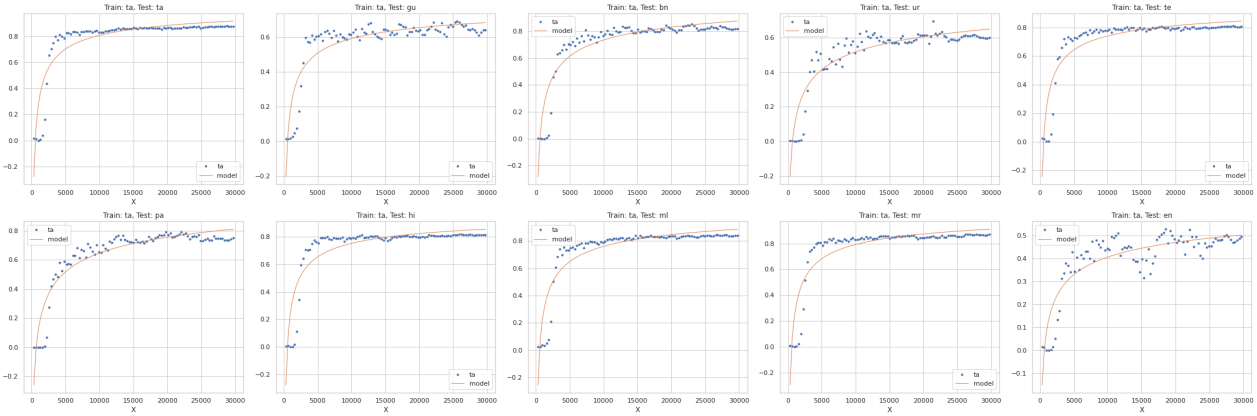

Figure 4: *Best-fit curves for MuRIL* when fine-tuned on Tamil for each of the target languages.

Test	Train	MuRIL		XLM-R	
		Edge Weight	R-squared	Edge Weight	R-squared
bn	bn	$1.2 - 29.0 * x^{-0.5}$	0.88	$1.3 - 11.5 * x^{-0.4}$	0.93
	en	$1.2 - 11.4 * x^{-0.4}$	0.78	$1.1 - 8.1 * x^{-0.3}$	0.89
	hi	$1.4 - 9.4 * x^{-0.3}$	0.85	$1.1 - 8.1 * x^{-0.3}$	0.92
	ml	$1.2 - 10.7 * x^{-0.3}$	0.86	$2.3 - 4.8 * x^{-0.1}$	0.92
	mr	$1.9 - 6.5 * x^{-0.2}$	0.88	$1.9 - 4.6 * x^{-0.1}$	0.93
	ta	$1.2 - 10.5 * x^{-0.3}$	0.83	$1.3 - 6.1 * x^{-0.2}$	0.90
	ur	$1.0 - 13.5 * x^{-0.4}$	0.88	$1.0 - 6.5 * x^{-0.3}$	0.91
en	bn	$0.9 - 4.4 * x^{-0.3}$	0.86	$1.0 - 5.2 * x^{-0.3}$	0.90
	en	$1.1 - 16.4 * x^{-0.4}$	0.82	$1.1 - 14.6 * x^{-0.4}$	0.85
	hi	$1.0 - 5.6 * x^{-0.3}$	0.88	$1.0 - 7.6 * x^{-0.3}$	0.90
	ml	$1.9 - 3.5 * x^{-0.1}$	0.88	$1.0 - 6.1 * x^{-0.3}$	0.86
	mr	$1.2 - 3.2 * x^{-0.2}$	0.84	$1.2 - 4.8 * x^{-0.2}$	0.91
	ta	$0.8 - 4.2 * x^{-0.3}$	0.76	$0.7 - 6.9 * x^{-0.4}$	0.76
	ur	-	0.88	$1.0 - 3.9 * x^{-0.2}$	0.90
gu	bn	$2.6 - 4.3 * x^{-0.1}$	0.93	$1.0 - 4.8 * x^{-0.3}$	0.88
	en	$0.9 - 5.5 * x^{-0.3}$	0.80	$0.7 - 11.3 * x^{-0.5}$	0.78
	hi	$1.2 - 5.4 * x^{-0.2}$	0.87	$0.7 - 13.3 * x^{-0.5}$	0.86
	ml	$1.2 - 7.8 * x^{-0.3}$	0.85	$1.6 - 4.3 * x^{-0.2}$	0.90
	mr	$1.1 - 6.4 * x^{-0.3}$	0.87	$1.1 - 6.0 * x^{-0.3}$	0.85
	ta	$0.8 - 11.3 * x^{-0.4}$	0.78	$1.0 - 7.6 * x^{-0.3}$	0.84
	ur	$1.4 - 3.4 * x^{-0.1}$	0.91	$1.6 - 3.2 * x^{-0.1}$	0.89
hi	bn	$0.9 - 17.2 * x^{-0.5}$	0.88	$1.2 - 4.8 * x^{-0.2}$	0.94
	en	$1.0 - 11.7 * x^{-0.4}$	0.83	$0.9 - 8.6 * x^{-0.4}$	0.88
	hi	$1.1 - 23.9 * x^{-0.5}$	0.90	$1.3 - 9.5 * x^{-0.3}$	0.92
	ml	$1.1 - 12.4 * x^{-0.4}$	0.85	$1.4 - 5.7 * x^{-0.2}$	0.90
	mr	$1.1 - 17.6 * x^{-0.5}$	0.85	$2.0 - 5.3 * x^{-0.2}$	0.93
	ta	$1.0 - 19.1 * x^{-0.5}$	0.78	$1.1 - 8.7 * x^{-0.3}$	0.88
	ur	$1.0 - 8.0 * x^{-0.3}$	0.92	$1.2 - 4.6 * x^{-0.2}$	0.94
ml	bn	$1.1 - 5.9 * x^{-0.3}$	0.88	$1.3 - 4.3 * x^{-0.2}$	0.92
	en	$1.2 - 5.0 * x^{-0.2}$	0.85	$0.8 - 7.6 * x^{-0.3}$	0.85
	hi	$2.1 - 5.0 * x^{-0.1}$	0.86	$1.0 - 10.8 * x^{-0.4}$	0.90
	ml	$1.1 - 21.4 * x^{-0.5}$	0.83	$1.3 - 7.8 * x^{-0.3}$	0.90
	mr	$1.5 - 7.3 * x^{-0.3}$	0.86	$1.4 - 6.4 * x^{-0.3}$	0.91
	ta	$1.1 - 12.8 * x^{-0.4}$	0.81	$1.1 - 10.0 * x^{-0.4}$	0.86
	ur	$1.1 - 5.1 * x^{-0.2}$	0.89	$1.1 - 4.7 * x^{-0.2}$	0.89
mr	bn	$1.0 - 9.3 * x^{-0.4}$	0.88	$1.1 - 5.1 * x^{-0.2}$	0.89
	en	$0.9 - 8.8 * x^{-0.3}$	0.81	$0.9 - 7.6 * x^{-0.3}$	0.87
	hi	$1.3 - 10.3 * x^{-0.3}$	0.86	$1.1 - 9.6 * x^{-0.4}$	0.91
	ml	$1.1 - 15.2 * x^{-0.4}$	0.83	$1.3 - 6.4 * x^{-0.3}$	0.90
	mr	$1.2 - 21.9 * x^{-0.5}$	0.86	$1.6 - 7.6 * x^{-0.3}$	0.92
	ta	$1.1 - 17.3 * x^{-0.4}$	0.79	$1.1 - 10.7 * x^{-0.4}$	0.85
	ur	$1.2 - 5.2 * x^{-0.2}$	0.92	$1.3 - 4.2 * x^{-0.2}$	0.91
pa	bn	$1.0 - 6.4 * x^{-0.3}$	0.86	$0.9 - 4.2 * x^{-0.3}$	0.82
	en	$1.2 - 4.0 * x^{-0.2}$	0.84	$1.1 - 2.9 * x^{-0.2}$	0.85
	hi	$1.9 - 5.9 * x^{-0.2}$	0.84	$1.8 - 4.0 * x^{-0.1}$	0.93
	ml	$1.0 - 9.7 * x^{-0.4}$	0.83	$1.2 - 3.6 * x^{-0.2}$	0.87
	mr	$2.7 - 5.3 * x^{-0.1}$	0.88	$1.3 - 3.9 * x^{-0.2}$	0.84
	ta	$1.4 - 6.3 * x^{-0.2}$	0.86	$1.0 - 4.7 * x^{-0.2}$	0.84
	ur	$1.2 - 4.5 * x^{-0.2}$	0.92	$0.8 - 4.2 * x^{-0.3}$	0.87
ta	bn	$1.1 - 7.2 * x^{-0.3}$	0.89	$1.0 - 4.6 * x^{-0.2}$	0.93
	en	$1.0 - 7.9 * x^{-0.3}$	0.83	$0.8 - 6.7 * x^{-0.3}$	0.86
	hi	$1.4 - 6.7 * x^{-0.3}$	0.90	$1.2 - 5.9 * x^{-0.3}$	0.92
	ml	$1.0 - 14.1 * x^{-0.4}$	0.83	$1.3 - 4.7 * x^{-0.2}$	0.92
	mr	$1.3 - 9.7 * x^{-0.3}$	0.86	$2.7 - 5.0 * x^{-0.1}$	0.94
	ta	$1.1 - 19.7 * x^{-0.5}$	0.79	$1.2 - 9.4 * x^{-0.3}$	0.88
	ur	$1.2 - 5.0 * x^{-0.2}$	0.92	$1.5 - 3.4 * x^{-0.1}$	0.92
te	bn	$1.1 - 5.4 * x^{-0.3}$	0.90	$0.8 - 4.7 * x^{-0.3}$	0.88
	en	$0.8 - 10.1 * x^{-0.4}$	0.79	$0.7 - 6.7 * x^{-0.4}$	0.83
	hi	$1.0 - 9.7 * x^{-0.4}$	0.91	$0.9 - 7.3 * x^{-0.3}$	0.86
	ml	$1.0 - 15.3 * x^{-0.4}$	0.83	$1.1 - 5.6 * x^{-0.3}$	0.88
	mr	$1.0 - 12.5 * x^{-0.4}$	0.87	$1.7 - 4.5 * x^{-0.2}$	0.93
	ta	$1.0 - 16.5 * x^{-0.4}$	0.81	$1.0 - 7.7 * x^{-0.3}$	0.87
	ur	$1.4 - 4.2 * x^{-0.2}$	0.91	$1.4 - 3.1 * x^{-0.1}$	0.90
ur	bn	$0.6 - 11.3 * x^{-0.5}$	0.86	-	0.76
	en	$1.1 - 5.5 * x^{-0.2}$	0.83	$1.0 - 5.9 * x^{-0.3}$	0.81
	hi	$2.6 - 4.8 * x^{-0.1}$	0.85	-	0.96
	ml	$1.1 - 8.2 * x^{-0.3}$	0.80	$2.5 - 5.0 * x^{-0.1}$	0.85
	mr	$4.3 - 6.1 * x^{-0.1}$	0.87	$5.2 - 7.0 * x^{-0.0}$	0.91
	ta	$1.1 - 5.0 * x^{-0.2}$	0.83	$1.2 - 5.2 * x^{-0.2}$	0.83
	ur	$1.0 - 42.2 * x^{-0.6}$	0.87	$1.1 - 20.9 * x^{-0.5}$	0.90

Table 11: *Power-law equations* empirically determined for each source and target pair. Please refer to Section A.5 for more details

Metric	Budget	Model	bn	en	hi	ml	mr	ta	ur
$GM_{\tau=0}$	1,000	$XLM\text{-}R_{large}$	128	157	145	134	133	163	140
		$MuRIL_{large}$	137	135	134	158	142	159	135
	5,000	$XLM\text{-}R_{large}$	704	792	693	794	696	628	693
		$MuRIL_{large}$	743	644	749	783	745	852	484
	10,000	$XLM\text{-}R_{large}$	1322	1349	1400	1481	1457	1479	1512
		$MuRIL_{large}$	1302	1468	1379	1421	1425	1448	1557
$GM_{\tau=1}$	1,000	$XLM\text{-}R_{large}$	126	160	159	134	129	163	129
		$MuRIL_{large}$	142	136	152	143	148	157	122
	5,000	$XLM\text{-}R_{large}$	710	805	713	803	707	639	623
		$MuRIL_{large}$	744	644	761	772	747	848	484
	10,000	$XLM\text{-}R_{large}$	1308	1363	1456	1465	1459	1471	1478
		$MuRIL_{large}$	1308	1488	1396	1406	1416	1441	1545

Table 12: *Optimal allocations under different budgets*. Please refer to Section A.5 for more details