Does Transliteration Help Multilingual Language Modeling?

Ibraheem Muhammad Moosa
Pennsylvania State University
ibraheem.moosa@psu.edu

Mahmud Elahi Akhter and **Ashfia Binte Habib**
Dept. of Electrical and Computer Engineering,
North South University, Dhaka, Bangladesh
{mahmud.akhter01,ashfia.habib}@northsouth.edu

Abstract

Script diversity presents a challenge to Multilingual Language Models (MLLM) by reducing lexical overlap among closely related languages. Therefore, transliterating closely related languages that use different writing scripts to a common script may improve the downstream task performance of MLLMs. We empirically measure the effect of transliteration on MLLMs in this context. We specifically focus on the Indic languages, which have the highest script diversity in the world, and we evaluate our models on the IndicGLUE benchmark. We perform the Mann-Whitney U test to rigorously verify whether the effect of transliteration is significant or not. We find that transliteration benefits the low-resource languages without negatively affecting the comparatively high-resource languages. We also measure the cross-lingual representation similarity of the models using centered kernel alignment on parallel sentences from the FLORES-101 dataset. We find that for parallel sentences across different languages, the transliteration-based model learns sentence representations that are more similar.

1 Introduction

In the last few years, we have seen impressive advances in many NLP tasks. These advances have been primarily led by the availability of large representative corpora and improvement in the architecture of large language models. While improving model architectures, training methods, regularization techniques, etc., can help advance the state of NLP in general, the unavailability of large, diverse corpora is the bottleneck for most languages (Joshi et al., 2020). Thus to inclusively advance the state of NLP across languages, it is crucial to develop techniques for training MLLMs that can extract the most out of existing multilingual corpora. Here, we focus on the issue of diverse writing scripts used by closely related languages that may prevent MLLMs from learning good cross-lingual representations. Previous papers (Pfeiffer et al., 2021) have noted that low-resource languages that use unique scripts tend to have very few tokens representing them at the tokenizer. As a result, these languages tend to have more *UNK*nown tokens, and the words in these languages tend to be more split up by subword tokenizers. Often we can easily transliterate from one script to another using rule-based systems. For example, there are established standards that can be used to transliterate Greek (ISO 843), Cyrillic (ISO 9), Indic scripts (ISO 15919), and Thai (ISO 11940) to the Latin script.

In this paper, we focus on the Indic languages, which have the highest script diversity in the world. Many South Asian and Southeast Asian languages are intimately connected linguistically, historically, phonologically (Littell et al., 2017) and phylogenetically. However, due to different scripts, it is difficult for MLLMs to fully exploit this shared information. Among the Indic languages we considered in this study we encounter eleven different scripts. These are shown in Table 1. Nevertheless, these scripts have shared ancestry from the ancient Brahmic script (Hockett et al., 1997; Coningham et al., 1996) and have similar structures that we can easily use to transliterate them to a common script. Also, many of these languages heavily borrow from Sanskrit, and due to its influence, many words are shared among these languages. Therefore, due to their relatedness and highly diverse script barrier, the Indic languages presents a unique opportunity to analyze the effects of transliteration on MLLMs.

We empirically measure the effect of transliteration on the downstream performance of MLLMs. We pretrain ALBERT (Base, 11M Parameters) (Lan et al., 2020) and RemBERT (Base, 192M Parameters) (Chung et al., 2020) models from scratch

Findings of the Association for Computational Linguistics: EACL 2023, pages 670–685

on Indic languages. We pretrain two variants of each model, one with the original writing scripts and the other after transliterating to a common writing script. Henceforth, we will refer to the transliterated script model as uni-script model and the other as a multi-script model. We evaluate the models on downstream tasks from the IndicGLUE benchmark dataset (Kakwani et al., 2020). In order to rigorously compare the two models, we finetune using nine random seeds on all downstream tasks. Then we perform the Mann-Whitney U test (MWU) between the uni-script and multi-script models. Using the MWU test, we conclude that transliteration significantly benefits the low-resource languages without negatively affecting the comparatively high-resource languages.

We also measure the Cross-Lingual Representation Similarity (CLRS) to understand why the uni-script model performs better than the multi-script model. To measure the CLRS, we use the centered kernel alignment (CKA) (Kornblith et al., 2019) similarity score. We measure the CKA similarity score between the hidden representations of the models on the parallel sentences of the Indic languages from the FLORES-101 dataset (Goyal et al., 2022). We find that, compared to the multi-script models, the uni-script models achieve a higher CKA score, and it is more stable throughout the hidden layers of the models. Based on this, we conclude that the uni-script models learn better cross-lingual representation than the multi-script models. In summary, our contributions are primarily three-fold:

1. We find that transliteration significantly benefits the low-resource languages without negatively affecting the comparatively high-resource languages.

2. We establish this finding through rigorous experiments and show the statistical significance along with the effect size of transliteration using the Mann-Whitney U test.

3. Using CKA on the FLORES-101 dataset, we show that transliteration helps MLLMs learn better cross-lingual representation.

Our code is available at Github[1] and our model weights can be downloaded from HF Hub [2] [3] [4] [5].

2 Motivation and Background

2.1 Motivation

In their study, Joshi et al. (2020) showed the resource disparity between low-resource and high-resource languages, and Ruder (2020) discussed the necessity of working with low-resource languages. A large body of work suggests that language-relatedness can help MLLMs achieve better performance on low-resource languages by leveraging related high-resource languages. For instance, Pires et al. (2019) found that lexical overlap improved mBERT's multilingual representation capability even though it learned to capture multilingual representations with zero lexical overlaps. Dabre et al. (2017) showed that transfer learning in the same or linguistically similar language family gives the best performance for NMT. Lauscher et al. (2020) found that language relatedness is crucial for POS-tagging and dependency parsing tasks. Although, corpus size is much more important for NLI and Question Answering tasks. Wu and Dredze (2020) showed that bilingual BERT outperformed monolingual BERT on low-resource languages when the languages were linguistically closely related. Nevertheless, mBERT outperformed bilingual BERT on low-resource languages.

2.2 Script Barrier in Multilingual Language Models

One of the major challenges in leveraging transfer between high-resource and low-resource languages is overcoming the script barrier. Script barrier exists when multiple closely related languages use different scripts. Anastasopoulos and Neubig (2019) found that for morphological inflection, script barrier between closely related languages impedes cross-lingual learning, and language relatedness improved cross-lingual transfer. Transliteration and phoneme-based techniques have been proposed to solve this issue. For example, Murikinati et al. (2020) expanded upon Anastasopoulos and Neubig (2019) and showed that both transliter-

[1] https://github.com/ibraheem-moosa/XLM-Indic

[2] https://huggingface.co/ibraheemmoosa/xlmindic-base-uniscript

[3] https://huggingface.co/ibraheemmoosa/xlmindic-base-multiscript

[4] https://huggingface.co/ibraheemmoosa/xlmindic-rembert-uniscript

[5] https://huggingface.co/ibraheemmoosa/xlmindic-rembert-multiscript

ation and grapheme to phoneme (g2p) conversion removes script barrier and improves cross-lingual morphological inflection and Rijhwani et al. (2019) showed that pivoting low-resource languages to their closely related high-resource languages results in better zero shot entity linking capacity and used phoneme-based pivoting to overcome the script barrier. Bharadwaj et al. (2016) showed that phoneme representation outperformed orthographic representations for NER. Chaudhary et al. (2018) also used phoneme representation to resolve script barriers and adapt word embeddings to low-resource languages.

2.3 Transliteration in Language Modeling

Different works have applied transliteration in different aspect for language models. For instance, Goyal et al. (2020) and Song et al. (2020) both utilized transliteration and showed that language relatedness was required for improving performance on NMT. Amrhein and Sennrich (2020) studied how transliteration improved NMT and came to the conclusion that transliteration offered significant improvement for low-resource languages with different scripts.

Khemchandani et al. (2021) showed on Indo-Aryan languages that language relatedness could be exploited through transliteration along with bilingual lexicon-based pseudo-translation and aligned loss to incorporate low-resource languages into pretrained mBERT. Muller et al. (2021) showed that for unseen languages, the script barrier hindered transfer between low-resource and high-resource languages for MLLMs and transliteration removed this barrier. They showed that transliterating Uyghur, Buryat, Erzya, Sorani, Meadow Mari, and Mingrelian to Latin script and finetuning mBERT on the respective corpus with masked language modeling objective improved their downstream POS performance significantly. In contrast, K et al. (2020) and Artetxe et al. (2020) proposes that mBERT can learn cross-lingual representations without any lexical overlap, a shared vocabulary, or joint training. However, these works focus on zero-shot cross-lingual transfer learning only. From the literature, it can be seen that many in the community believe transliteration to be a potential solution for script barriers. However, most of the work shows the benefits of transliteration for NMT. Nevertheless, there is no solid empirical analysis of the effects of transliteration for MLLMs apart from Dhamecha et al. (2021); Muller et al. (2021). Hence, the motivation behind this paper is to provide a solid empirical analysis of the effect of transliteration for MLLMs with statistical analysis and determine whether or not it helps models learn better cross-lingual representation.

It should also be noted that, even though our idea seems to be similar to Muller et al. (2021) and Dhamecha et al. (2021), there are major differences. For instance, Muller et al. (2021) adapted existing pretrained model to very low-resource languages. Whereas, we focus on training the models with transliteration from scratch. We also train our models on 20 languages and evaluate on more than 50 tasks. Unlike Dhamecha et al. (2021), we also include Dravidian Languages in our analysis. Furthermore, we focused on the issue of script barrier while Dhamecha et al. (2021) focused on multilingual fine-tuning. Whereas, we adopt multilingual fine-tuning on all our models. Thus the improvement we see comes only from circumventing the script barrier. Moreover, we have provided statistical testing to show the significance of transliteration instead of just showing better metrics. We also performed cross-lingual representation similarity analysis to show the benefits of transliteration.

2.4 Cross Lingual Similarity Learning in Language Modeling

Several techniques have recently been used to study the hidden representations of multilingual language models. Kudugunta et al. (2019) study CLRS of NMT models using SVCCA (Raghu et al., 2017). Singh et al. (2019) used PWCCA (Morcos et al., 2018) to study the CLRS of mBERT and found that it drastically fell with depth. (Conneau et al., 2020) have used CKA to study the CLRS of bilingual BERT models. They found that similarity is highest in the first few layers and drops moderately with depth. Müller et al. (2021) used CKA to study CLRS of mBERT before and after finetuning on downstream tasks. They found in all cases that CLRS increases steadily in the first five layers, then it decreases in the later layers. From this, they concluded that mBERT learns multilingual alignment in the early layers and preserves it throughout finetuning. Del and Fishel (2021) applied various similarity measures to understand CLRS of various multilingual masked language models. Their results also show that CLRS increases in the first half of the models, while in the later layers, this

similarity steadily falls.

3 Experiment and Results

3.1 Mann–Whitney U test

We perform Mann–Whitney U test (MWU) (Mann and Whitney, 1947; Wilcoxon, 1945) to determine if the performance differences between the multi-script and the uni-script models are significant. In short, it tells us the effect of transliteration on model performance. MWU is a non-parametric hypothesis test between two groups/populations. MWU is chosen because it has weak assumptions. The only assumptions of MWU are that the samples of the two groups are independent of each other, and the samples are ordinal. Under the MWU, our null hypothesis or $\mathbf{h_0}$ is that the performances of the uni-script (group 1) and the multi-script (group 2) models are similar, and the alternative hypothesis or $\mathbf{h_a}$ is that the performances (groups) are different. We set our confidence interval α at 0.05 and reject the $\mathbf{h_0}$ for the p-values $< \alpha$. We also report three test statistics as the p-value only gives statistical significance, which can be misleading at times (Sullivan and Feinn, 2012).

The test statistics are three different effect sizes that convey three different information. These test statistics are absolute effect size (δ), common language effect size (ρ), and standardized effect size (r). The absolute effect size δ is the difference between the mean of the models' performance metric, which is given as,

$$\delta = \mu_{\text{uni-script}} - \mu_{\text{multi-script}}$$

for any given task and language. When the $\mathbf{h_0}$ is rejected for any given task, a positive δ indicates the uni-script model is better, and a negative δ indicates the multi-script model is better. The details and results of common language effect size (ρ), and standardized effect size (r) are presented in appendix D.

3.2 Dataset

The ALBERT models were pretrained on a subset of the OSCAR corpus containing Indo-Aryan languages. We use the unshuffled deduplicated version of OSCAR corpus (Ortiz Su'arez et al., 2019) available via Huggingface datasets library (Lhoest et al., 2021). We pretrain on Panjabi, Hindi, Bengali, Oriya, Assamese, Gujarati, Marathi, Sinhala, Nepali, Sanskrit, Goan Konkani, Maithili, Bihari, and Bishnupriya portion of the OSCAR corpus.

The RemBERT models were trained on a significantly larger pretraining corpus with additional languages. We pretrained the RemBERT models on a combination of Wikipedia (Foundation), mC4 (Raffel et al., 2019), OSCAR2109 (Abadji et al., 2021) and OSCAR corpus. These datasets are also available via the Huggingface datasets library. In addition to the languages in the ALBERT pretraining corpus, we consider English, four Dravidian languages Kannada, Telugu, Malayalam, and Tamil, and an Indo-Aryan language Dhivehi. We evaluate the models on four downstream tasks from IndicGLUE (Kakwani et al., 2020), which are News Article Classification, WSTP, CSQA, and NER. We use the balanced Wikiann dataset from Rahimi et al. (2019) for NER. In addition, we evaluate the models on other publicly available datasets that are part of the IndicGLUE benchmark. These are BBC Hindi News Classification, Soham Bengali News Classification, INLTK Headlines Classification, IITP Movie, and Product Review Sentiment Analysis (Akhtar et al., 2016), MIDAS Discourse Mode Classification (Dhanwal et al., 2020) and ACTSA Sentiment Classification (Mukku and Mamidi, 2017) datasets.

Lang.	Sub-family	Script	Size(GB)
en	Germanic	Latin	131
hi	Central Indo-Aryan	Devanagari	43
mr	Southern Indo-Aryan	Devanagari	35
bn	Eastern Indo-Aryan	Bengali	28
ta	South Dravidian	Tamil	22
ml	South Dravidian	Malayalam	10
te	South-Central Dravidian	Telugu	7
kn	South Dravidian	Kannada	6
si	Insular Indo-Aryan	Sinhala	5
ne	Northern Indo-Aryan	Devanagari	4
gu	Western Indo-Aryan	Gujarati	3.5
pa	Northwestern Indo-Aryan	Gurmukhi	2
or	Eastern Indo-Aryan	Oriya	0.5
sa	Sanskrit	Devanagari	0.2
as	Eastern Indo-Aryan	Bengali	0.1
dv	Insular Indo-Aryan	Thaana	0.1
bpy	Eastern Indo-Aryan	Bengali	< 0.1
gom	Southern Indo-Aryan	Devanagari	< 0.1
bh	Eastern Indo-Aryan	Devanagari	< 0.1
mai	Eastern Indo-Aryan	Devanagari	< 0.1

Table 1: Languages in our pretraining corpus and their writing scripts and the pretraining corpus sizes used for the RemBERT model

3.3 Transliteration Method

We transliterate Indic language texts to Latin script using the ISO 15919 transliteration scheme. We tested with two publicly available implementations of this scheme, Aksharamukha (Rajan, 2015) and PyICU (PyICU). We found the quality of translit-

eration of the Aksharamukha library to be better. Thus we use this library for transliterating the inputs to the ALBERT uni-script model. However, the Aksharamukha implementation is very slow compared to the PyICU implementation. As we significantly expanded our pretraining corpus for the RemBERT model, we switched to PyICU for the RemBERT uni-script model.

3.4 Downstream Finetuning

We finetune the models on each downstream task independently. The specific hyperparameters used for each task are reported in the appendix B. On all tasks, we finetune with nine random seeds and report the average and standard deviation of the metrics. In Table 2 and Table 4, we report the performances on IndicGLUE benchmark tasks and in Table 3 on other publicly available datasets. Here, we discuss the results on each of the the models on each of the tasks. Furthermore, in appendix D, we show the test statistics for all the datasets.

Wikipedia Section Title Prediction: For both RemBERT and ALBERT models, the uni-script model performed better on all languages except Malayalam (ml). We noticed that a letter of Malayalam script is not properly transliterated by the PyICU library. This introduced some artifacts in the form of unnecessary splitting of words by the subword tokenizer.

News Category Classification: It is interesting that on this task the uni-script models performed better for Panjabi (pa) and Oriya (or) languages. It is clear from Table 1 that these two languages are low-resource compared to Bengali (bn) and Marathi (mr). On Bengali and Marathi we see slight performance degradation which is not statistically significant. This shows the validity of our first finding.

Named Entity Recognition: On this task we see that the uni-script model performs much better for Assamese (as), Oriya(or), Panjabi (pa) and Gujarati (gu). These languages are low-resource and here again the uni-script model shines. The large performance improvement on this task can be explained by the fact that Named Entities usually have the same spelling after transliteration for Indian languages. Thus the uni-script model has better chances for learning various named-entities during pre-training.

Article Genre, Sentiment & Discourse Mode Classification: We evaluate the models on various other sequence classification datasets that are part of the IndicGLUE benchmark. Here again the uni-script model usually performs better than the multi-script model. However for two tasks in Malayalam (ml) and Tamil (ta) we see better performance for the multi-script model. We already mentioned that there is some tokenization issue for Malayalam which can explain the results for Malayalam. The results for Tamil suggests that it may be a good idea to try both uni-script and multi-script model if they are available to see which performs best on a particular task. However this is the only instance of a task where we see the multi-script model perform better.

3.5 Zero Shot Capability Testing

We use the CSQA task to test the zero-shot capability of the models as we can use the models without finetuning. This task is designed to test whether language models can be used as knowledge bases (Petroni et al., 2019). In Table 4 we report the results. We note that the RemBERT models perform much better than the ALBERT models on this task. This is expected as the ALBERT models' memorization capability is hampered by weight sharing.

The ALBERT uni-script model is better on all languages compared to the ALBERT multi-script model. This shows the potential of a uni-script model in a restricted low parameter situation. For the RemBERT models, the results are mixed. However, on average the uni-script model performs better than the multi-script model. The worst results are for Malayalam (ml) which as we mentiond before has some tokenization issues.

4 Cross-lingual Representation Similarity

In this section, we analyze why the uni-script model performs better than the multi-script model from the perspective of Cross-Lingual Representation Similarity. Following (Müller et al., 2021), (Conneau et al., 2020) and (Del and Fishel, 2021) we apply CKA to measure CLRS. We use the CKA implementation from the Ecco library (Alammar, 2021). We use parallel sentences on thirteen languages from the FLORES-101 (Goyal et al., 2022) dataset. For the ALBERT models, which are trained on only the Indo-Aryan languages, we only consider Panjabi, Hindi, Bengali, Oriya, Assamese, Gujarati, Marathi, and Nepali sentences. For the RemBERT models, we additionally consider Kannada, Telugu, Malayalam, Tamil, and English sentences.

Model	pa	hi	bn	or	as	gu	mr	kn	te	ml	ta	avg
Wikipedia Section Title Prediction												
RemBERT$_{MS}$	68.42±0.92	70.90±0.39	72.58±0.45	69.92±0.90	68.37±1.37	72.93±0.58	73.23±0.61	71.67±0.41	92.98±0.19	69.03±0.57	69.77±0.45	73.00
RemBERT$_{US}$	71.01±0.22	72.45±0.29	73.65±0.21	75.37±0.69	72.50±0.91	76.35±0.29	74.58±0.72	74.21±0.29	93.66±0.09	69.33±0.35	70.63±0.22	74.89
δ	**2.59**	**1.55**	**1.07**	**5.45**	**4.13**	**3.42**	**1.34**	**2.54**	**0.68**	**0.31**	**0.86**	**1.89**
$p-value$	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004	0.0035	0.0004	0.0004	0.2505	0.0006	-
ALBERT$_{MS}$	74.33±0.83	78.18±0.33	81.18±0.28	74.35±1.2	76.70±0.83	76.37±0.53	79.10±0.84	-	-	-	-	77.17
ALBERT$_{US}$	77.55±0.61	82.24±0.18	84.38±0.29	81.47±0.99	81.74±0.82	82.39±0.27	82.74±0.52	-	-	-	-	81.78
δ	**3.22**	**4.06**	**3.20**	**7.12**	**5.04**	**6.02**	**3.64**	-	-	-	-	**4.61**
$p-value$	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004	-	-	-	-	-
News Category Classification												
RemBERT$_{MS}$	95.67±0.38	-	97.90±0.17	96.59±0.18	-	98.22±0.58	99.16±0.16	97.23±0.10	99.03±0.12	91.25±0.43	97.33±0.18	96.93
RemBERT$_{US}$	96.92±0.29	-	97.78±0.12	97.55±0.14	-	99.02±0.14	99.14±0.21	97.10±0.12	99.03±0.66	92.08±0.40	97.49±0.20	97.34
δ	**1.24**	-	**-0.11**	**0.95**	-	**0.80**	**-0.03**	**-0.13**	**0.00**	**0.83**	**0.16**	**0.41**
$p-value$	0.0003	-	0.0981	0.0004	-	0.0040	0.7783	0.0995	0.7548	0.0014	0.0814	-
ALBERT$_{MS}$	96.83±0.19	-	98.14±0.14	98.09±0.16	-	98.80±0.43	99.58±0.25	-	-	-	-	98.30
ALBERT$_{US}$	97.90±0.17	-	97.99±0.22	98.77±0.12	-	99.40±0.54	99.47±0.21	-	-	-	-	98.70
δ	**1.07**	-	**-0.15**	**0.68**	-	**0.60**	**-0.18**	-	-	-	-	**0.40**
$p-value$	0.0003	-	0.181	0.0004	-	0.03084	0.1683	-	-	-	-	-
Named Entity Recognition (F1-Score)												
RemBERT$_{MS}$	69.47±1.72	90.95±0.33	95.51±0.18	87.92±1.26	79±0.22	69±0.94	90.72±0.17	72.65±1.81	81.82±1.81	89.17±0.25	90.07±0.33	83.40
RemBERT$_{US}$	81.91±1.93	91.73±0.39	96.19±0.21	88.92±2.88	83.50±2.75	80.25±1.42	90.75±0.35	78.98±1.50	84.97±0.45	89.26±0.46	90.18±0.27	86.97
δ	**12.44**	**0.78**	**0.68**	**1.00**	**4.28**	**10.31**	**0.02**	**6.33**	**3.15**	**0.01**	**0.12**	**3.56**
$p-value$	0.00004	0.0005	0.00001	0.1615	0.0019	0.00004	0.6665	0.00004	0.00004	0.7304	0.2973	-
ALBERT$_{MS}$	76.69±1.5	91.80±0.42	96.39±0.19	84.18±1.8	75.45±1.8	69.10±2.9	88.72±0.40	-	-	-	-	83.19
ALBERT$_{US}$	85.42±1.9	92.93±0.21	97.31±0.22	93.54±0.58	89.06±2.2	80.16±0.15	90.56±0.44	-	-	-	-	89.85
δ	**8.73**	**1.13**	**0.92**	**9.36**	**13.61**	**11.06**	**1.84**	-	-	-	-	**6.66**
$p-value$	0.0004066	0.0004066	0.0003983	0.0004038	0.000401	0.0004066	0.0004095	-	-	-	-	-

orange indicates the multi-script and uni-script models are equal and blue indicates the uni-script model is better

Table 2: Results on Classification Tasks from IndicGLUE Benchmark

Language	Dataset	RemBERT$_{MS}$	RemBERT$_{US}$	δ	$p-value$	ALBERT$_{MS}$	ALBERT$_{US}$	δ	$p-value$
Article Genre Classification									
hi	BBC News	76.80±0.84	77.78±0.92	**0.98**	0.0466	77.28±1.51	79.14±0.60	**1.86**	0.0088
bn	Soham News Article Classification	92.86±0.10	93.69±0.20	**0.83**	0.0004	93.22±0.49	93.89±0.48	**0.67**	0.0090
gu	INLTK Headlines	90.27±0.47	91.60±0.28	**1.33**	0.0004	90.41±0.69	90.73±0.75	**0.32**	0.6249
mr	INLTK Headlines	91.24±0.50	92.27±0.39	**1.03**	0.0008	92.21±0.23	92.04±0.47	**-0.17**	0.3503
ml	INLTK Headlines	94.11±0.49	93.33±0.22	**-0.78**	0.003	-	-	-	-
ta	INLTK Headlines	95.59±0.70	94.93±0.30	**-0.65**	0.013	-	-	-	-
Sentiment Analysis									
hi	IITP Product Reviews	72.17±1.98	72.85±0.63	**0.68**	0.9646	76.33±0.84	77.18±0.77	**0.85**	0.04099
hi	IITP Movie Reviews	58.66±1.09	62.65±2.74	**3.99**	0.0023	65.91±2.2	66.34±0.16	**0.15**	0.8941
te	ACTSA	61.18±1.38	60.53±0.85	**-0.66**	0.1981	-	-	-	-
Discourse Mode Classification									
hi	MIDAS Discourse	78.07±0.83	79.46±0.67	**1.39**	0.0415	78.39±0.33	78.54±0.91	**0.15**	0.7561

orange indicates the multi-script and uni-script models are equal, cyan indicates multi-script is better than uni-script models and blue indicates vice versa

Table 3: Accuracy on Public Datasets

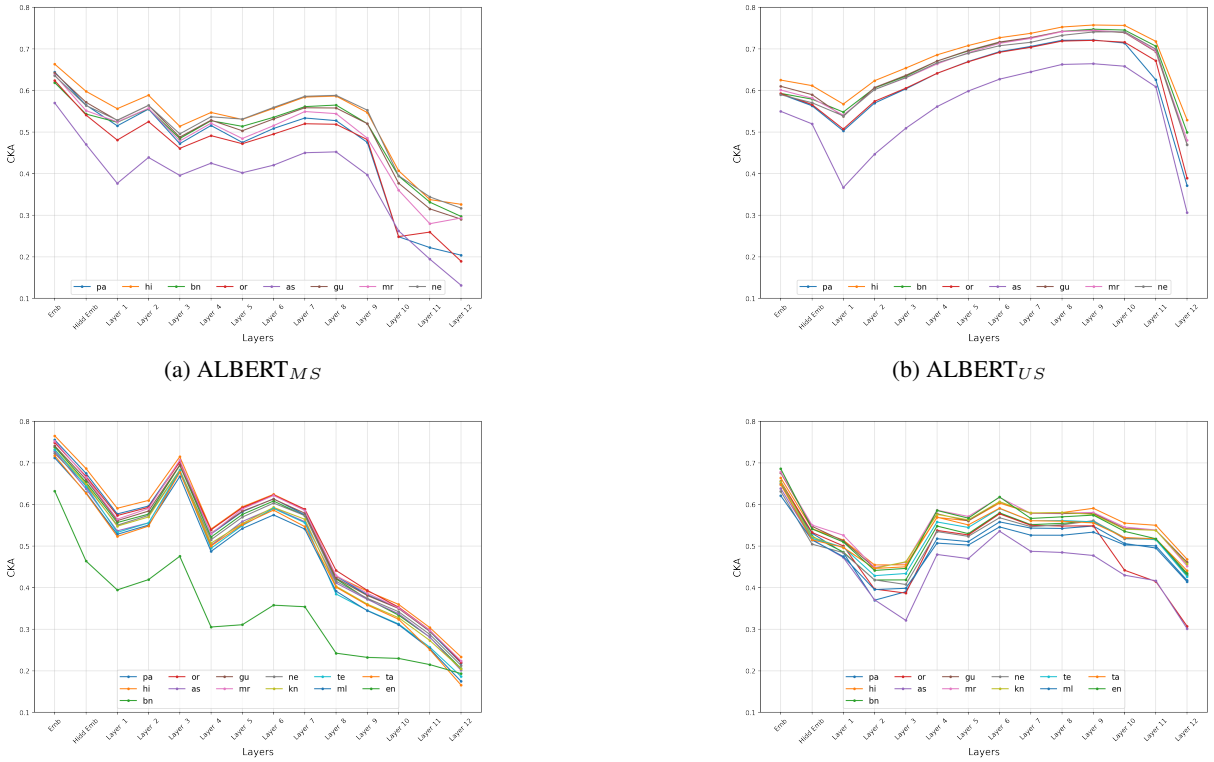
Model	pa	hi	bn	or	as	gu	mr	ta	te	ml	kn	avg
Cloze-style QA (Zero Shot)												
RemBERT$_{MS}$	33.93	39.06	38.93	37.32	37.66	84.21	46.15	37.02	34.42	38.45	40.75	42.53
RemBERT$_{US}$	33.92	40.10	39.62	38.28	39.26	85.37	45.92	36.68	34.36	37.16	44.29	43.17
δ	**-0.01**	**1.04**	**0.69**	**0.96**	**1.6**	**1.16**	**-0.23**	**-0.34**	**-0.06**	**-1.29**	**3.54**	**0.64**
ALBERT$_{MS}$	31.04	36.72	35.19	34.63	33.92	59.86	36.14	-	-	-	-	38.21
ALBERT$_{US}$	32.77	38.52	36.38	36.00	37.36	70.22	39.53	-	-	-	-	41.54
δ	**1.73**	**1.8**	**1.19**	**1.37**	**3.44**	**10.36**	**3.39**	-	-	-	-	**3.33**

cyan indicates multi-script is better than uni-script models and blue indicates vice versa

Table 4: Test accuracy on CSQA

First, we calculate the sentence embeddings of these parallel sentences from the models. Sentence embedding is calculated by averaging the hidden state representations of the tokens. Then, we calculate the CKA similarity score between the sentence embeddings for each language pair. For each language, we average its CKA similarity scores. In Figure 1 we plot this average CKA similarity for each layer of the models.

We see that CLRS score drops significantly at the last layer for all models. However, the uni-script models retain high CLRS score until the eleventh layer, whereas the multi-script models have low CLRS score from the ninth layer. Overall the CLRS score of the uni-script models are more stable. This indicates that the uni-script models have learned



(a) ALBERT$_{MS}$ (b) ALBERT$_{US}$

(c) RemBERT$_{MS}$ (d) RemBERT$_{US}$

Figure 1: CKA Similarity Score for the multi-script and uni-script models

better cross-lingual representations.

5 Tokenizer Quality Analysis

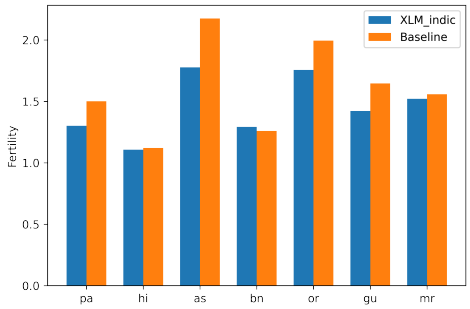
In terms of performance, we expect the transliteration model to exploit better tokenization across the languages. Following (Ács, 2019) and (Rust et al., 2021), we measure the subword fertility (average number of tokens per word) and the ratio of words unbroken by the tokenizer. From figure 2, we can see that transliteration reduces the splitting of words. This indicates that many words that were represented by different tokens in the multi-script model are represented by a single token in the transliteration model. On average, the ALBERT uni-script tokenizer has a lower subword fertility score of 1.55 compared to the multi-script tokenizer's 1.825. The uni-scirpt tokenizer also has a lower proportion of continued word score of 0.36 while the multi-script tokenizer has a score of 0.45.

6 Conclusion and Future Work

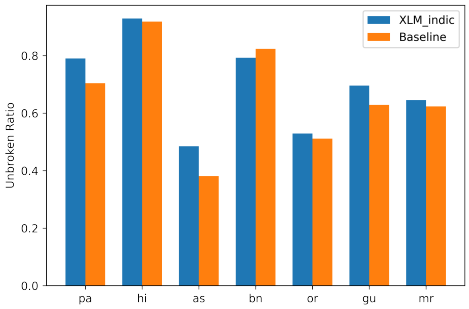
In this paper, we show that transliterating closely related languages to a common script improves multilingual language model performance and leads to better cross-lingual representations. We conducted rigorous statistical analysis to quantify the significance and effect size of transliteration on downstream task performance. We found that transliteration especially improves performance on comparatively low-resource languages and did not hurt the performance on high-resource languages. This findings are in agreement with (Dhamecha et al., 2021; Muller et al., 2021). Our results indicate that in other scenarios where closely related languages use different scripts, transliteration can be used to improve the performance of language models. For example, Slavic and Turkic languages present similar scenarios. We would like to extend our study to models at different scales and more languages in the future. Also, another interesting future direction would be to just use the transliteration for pretraining signal but give the model the ability to deal with the original scripts.

Limitations

A limitation of our work is that it introduces a transliteration step into the model pipeline. Thus we need a stable implementation of the transliteration scheme. Thus the model can become tied to a specific version of the transliteration library. Also



(a) Subword Fertility.



(b) Unbroken Ratio.

Figure 2: Subword fertility (lower is better) and unbroken ratio (higher is better)

the transliteration scheme is not perfect as we saw for Malayalam, it introduced some artifacts. Finally given our limited computational budget, we could not run experiments with a lot of models at different scales. Thus the impact of transliteration over different model scales has not been explored. Even though our work has these limitations, it clearly shows transliteration as an important tool for training better multilingual models.

Ethics Statement

In their study, Joshi et al. (2020) showed the resource disparity between low-resource and high-resource languages, and (Ruder, 2020) also highlighted the necessity of working with low-resource languages. However, creating representative and inclusive corpora is a difficult task and an ongoing process and is not always possible for many low-resource languages. Thus to inclusively advance the state of NLP across languages, it is crucial to develop techniques for training MLLMs that can extract the most out of existing multilingual corpora. Hence, we believe our analysis might help MLLMS with low-resource languages in real-world applications. However, there is one ethical issue that we want to state explicitly. Even though we pretrain on a comparatively large multilingual corpus, the model may exhibit harmful gender, ethnic and political bias. If the model is fine-tuned on a task where these issues are important, it is necessary to take special consideration when relying on the model's decisions.

References

Julien Abadji, Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2021. Ungoliant: An optimized pipeline for the generation of a very large-scale multilingual web corpus. In *CMLC-9*, Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-9) 2021. Limerick, 12 July 2021 (Online-Event), pages 1 – 9, Mannheim. Leibniz-Institut für Deutsche Sprache.

Judit Ács. 2019. Exploring BERT's Vocabulary. *Blog Post*.

Md. Shad Akhtar, Asif Ekbal, and Pushpak Bhattacharyya. 2016. Aspect based sentiment analysis: Category detection and sentiment classification for hindi. In *proceedings of the 17th International Conference on Intelligent Text Processing and Computational Linguistics (CICLING 2016), April 3-9, 2016, Konya, Turkey*, pages 246–257. Association for Computational Linguistics.

J Alammar. 2021. Ecco: An open source library for the explainability of transformer language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 249–257, Online. Association for Computational Linguistics.

Chantal Amrhein and Rico Sennrich. 2020. On Romanization for model transfer between scripts in neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2461–2469, Online. Association for Computational Linguistics.

Antonios Anastasopoulos and Graham Neubig. 2019. Pushing the limits of low-resource morphological inflection. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 984–996, Hong Kong, China. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Akash Bharadwaj, David Mortensen, Chris Dyer, and Jaime Carbonell. 2016. Phonologically aware neural model for named entity recognition in low resource transfer settings. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1462–1472, Austin, Texas. Association for Computational Linguistics.

Aditi Chaudhary, Chunting Zhou, Lori Levin, Graham Neubig, David R. Mortensen, and Jaime Carbonell. 2018. Adapting word embeddings to new languages with morphological and phonological subword representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3285–3295, Brussels, Belgium. Association for Computational Linguistics.

Hyung Won Chung, Thibault Févry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2020. Rethinking embedding coupling in pre-trained language models. *CoRR*, abs/2010.12821.

R.A.E. Coningham, F.R. Allchin, C.M. Batt, and D. Lucy. 1996. Passage to india? anuradhapura and the early use of the brahmi script. *Cambridge Archaeological Journal*, 6(1):73–97.

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 6022–6034. Association for Computational Linguistics.

Raj Dabre, Tetsuji Nakagawa, and Hideto Kazawa. 2017. An empirical study of language relatedness for transfer learning in neural machine translation. In *Proceedings of the 31st Pacific Asia Conference on Language, Information and Computation*, pages 282–286. The National University (Phillippines).

Maksym Del and Mark Fishel. 2021. Establishing interlingua in multilingual language models. *CoRR*, abs/2109.01207.

Tejas Dhamecha, Rudra Murthy, Samarth Bharadwaj, Karthik Sankaranarayanan, and Pushpak Bhattacharyya. 2021. Role of Language Relatedness in Multilingual Fine-tuning of Language Models: A Case Study in Indo-Aryan Languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8584–8595, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Swapnil Dhanwal, Hritwik Dutta, Hitesh Nankani, Nilay Shrivastava, Yaman Kumar, Junyi Jessy Li, Debanjan Mahata, Rakesh Gosangi, Haimin Zhang, Rajiv Ratn Shah, and Amanda Stent. 2020. An annotated dataset of discourse modes in Hindi stories. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1191–1196, Marseille, France. European Language Resources Association.

Wikimedia Foundation. Wikimedia downloads.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Vikrant Goyal, Sourav Kumar, and Dipti Misra Sharma. 2020. Efficient neural machine translation for low-resource languages via exploiting related languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 162–168, Online. Association for Computational Linguistics.

Charles F. Hockett, Peter T. Daniels, and William Bright. 1997. The world's writing systems. *Language*, 73(2):379.

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 the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Karthikeyan K, Zihan Wang, Stephen Mayhew, and Dan Roth. 2020. Cross-lingual ability of multilingual BERT: an empirical study. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961, Online. Association for Computational Linguistics.

Yash Khemchandani, Sarvesh Mehtani, Vaidehi Patil, Abhijeet Awasthi, Partha Talukdar, and Sunita Sarawagi. 2021. Exploiting language relatedness for low web-resource language model adaptation: An Indic languages study. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1312–1323, Online. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey E. Hinton. 2019. Similarity of neural network representations revisited. *CoRR*, abs/1905.00414.

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

Sneha Kudugunta, Ankur Bapna, Isaac Caswell, and Orhan Firat. 2019. Investigating multilingual NMT representations at scale. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1565–1575, Hong Kong, China. Association for Computational Linguistics.

Anoop Kunchukuttan. 2020. The IndicNLP Library. https://github.com/anoopkunchukuttan/indic_nlp_library/blob/master/docs/indicnlp.pdf.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online. Association for Computational Linguistics.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander Rush, and Thomas Wolf. 2021. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Patrick Littell, David R. Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. URIEL and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 8–14, Valencia, Spain. Association for Computational Linguistics.

Daniel Lüdecke. 2020. *sjstats: Statistical Functions for Regression Models (Version 0.18.0)*.

H. B. Mann and D. R. Whitney. 1947. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, 18(1):50 – 60.

Ari S. Morcos, Maithra Raghu, and Samy Bengio. 2018. Insights on representational similarity in neural networks with canonical correlation. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 5732–5741.

Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2021. On the stability of fine-tuning BERT: misconceptions, explanations, and strong baselines. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Sandeep Sricharan Mukku and Radhika Mamidi. 2017. ACTSA: Annotated corpus for Telugu sentiment analysis. In *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 54–58, Copenhagen, Denmark. Association for Computational Linguistics.

Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2021. When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462, Online. Association for Computational Linguistics.

Benjamin Müller, Yanai Elazar, Benoît Sagot, and Djamé Seddah. 2021. First align, then predict: Understanding the cross-lingual ability of multilingual BERT. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 2214–2231. Association for Computational Linguistics.

Nikitha Murikinati, Antonios Anastasopoulos, and Graham Neubig. 2020. Transliteration for cross-lingual morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 189–197, Online. Association for Computational Linguistics.

Pedro Javier Ortiz Su'arez, Benoit Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. In *Proceedings of the Workshop on Challenges in the Management of Large Corpora*

(CMLC-7) 2019, Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, pages 9 – 16, Mannheim. Leibniz-Institut f"ur Deutsche Sprache.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulic, Iryna Gurevych, and Sebastian Ruder. 2021. Unks everywhere: Adapting multilingual language models to new scripts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 10186–10203. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

PyICU. Pyicu transliteration tool. https://pypi.org/project/PyICU/. Accessed: 2022-03-15.

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

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. 2017. SVCCA: singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6076–6085.

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, Florence, Italy. Association for Computational Linguistics.

Vinodh Rajan. 2015. Aksharamukha transliteration tool. https://github.com/virtualvinodh/aksharamukha-python. Accessed: 2021-10-04.

Shruti Rijhwani, Jiateng Xie, Graham Neubig, and Jaime G. Carbonell. 2019. Zero-shot neural transfer for cross-lingual entity linking. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6924–6931. AAAI Press.

Sebastian Ruder. 2020. Why You Should Do NLP Beyond English. http://ruder.io/nlp-beyond-english.

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

Jasdeep Singh, Bryan McCann, Richard Socher, and Caiming Xiong. 2019. BERT is not an interlingua and the bias of tokenization. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP, DeepLo@EMNLP-IJCNLP 2019, Hong Kong, China, November 3, 2019*, pages 47–55. Association for Computational Linguistics.

Haiyue Song, Raj Dabre, Zhuoyuan Mao, Fei Cheng, Sadao Kurohashi, and Eiichiro Sumita. 2020. Pre-training via leveraging assisting languages for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 279–285, Online. Association for Computational Linguistics.

G. M. Sullivan and R. Feinn. 2012. Using Effect Size-or Why the P Value Is Not Enough. *J Grad Med Educ*, 4(3):279–282.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu,

Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.

A Cloze Style QA Evaluation Method

Since a word can be tokenized to multiple tokens by the subword tokenizer, correctly evaluating the model on this task requires special care. Specifically, we have to use the same number of mask tokens as the number of subword tokens that a word gets split into. Then we calculate the probability for the word by multiplying the probability of the subword tokens predicted by the masked language model.

B Pretraining Details

Corpus Preparation: Since the OSCAR corpus contains raw text from the Web, we apply a few filtering and normalization. First, we discard entries where the dominant script does not match the language tag provided by the OSCAR corpus. Then we use the IndicNLP normalizer (Kunchukuttan, 2020) to normalize the raw text. For the uniscript model, we then transliterate all the text to ISO-15919 format using the Aksharamukha (Rajan, 2015) library.

For the RemBERT models we do not perform any of the filtering mentioned above since our pretraining corpus is comparatively very large. In this case, we use the PyICU library (PyICU) for transliterating to ISO-15919 format.

Tokenizer Training: For the ALBERT models, we train two SentencePiece tokenizers (Kudo and Richardson, 2018) on the transliterated and the non-transliterated corpus with a vocabulary size of 50,000. For the RemBERT models we train Unigram tokenizers from the Tokenizers library (Wolf et al., 2020) with a vocabulary size of 65,536.

ALBERT Model Training: We first pretrained an ALBERT base model from scratch on the non-transliterated corpus as our baseline. Afterward, we pretrained another ALBERT base from scratch on the transliterated corpus. We chose the base model due to computing constraints. We trained the models on a single TPUv3 VM. Both models were trained using the same hyperparameters. We followed the hyperparameters used in (Lan et al., 2020) except for batch size and learning rate. The pretraining objective is also the same as (Lan et al., 2020).We used a batch size of 256, which is the highest that fits into TPU memory, whereas the ALBERT paper used a batch size of 4096. As our batch size is 1/16th of the ALBERT paper, we use a learning rate of 1e-3/8, which is approximately 1/16th of the learning rate used in the ALBERT paper (1.76e-2). Additionally, we use the Adam optimizer (Kingma and Ba, 2015) instead of the LAMB optimizer. The rest of the hyperparameters were the same as the ALBERT paper. Specifically, we use a sequence length of 512 with absolute positional encoding, weight decay of 1e-2, warmup steps of 5000, max gradient norm of 1.0, and Adam epsilon of 1e-6. The models were trained for 1M steps. Each model took about 7.5 days to train. We use the ALBERT implementation from the Huggingface Transformers Library (Wolf et al., 2020).

RemBERT Model Training: We pretrained an RemBERT base models similar to the ALBERT models. We trained the models on a single TPUv3 VM. Both models were trained using the same hyperparameters. We followed the hyperparameters used in (Chung et al., 2020) except for batch size and learning rate. The pretraining objective is also the same as (Chung et al., 2020). We used a batch size of 256, which is the highest that fits into TPU memory, whereas the RemBERT paper used a batch size of 2048. As our batch size is 1/8th of the RemBERT paper, we use a learning rate of 2e-4/8, which is 1/8th of the learning rate used in the RemBERT paper. Similar to the ALBERT model, we use the Adam optimizer (Kingma and Ba, 2015). The rest of the hyperparameters were the same as the RemBERT paper. Specifically, we use a sequence length of 512 with absolute positional encoding, weight decay of 1e-2, warmup steps of 15000, max gradient norm of 1.0, and Adam epsilon of 1e-6. The models were trained for 1M steps. Each model took about 7.5 days to train. We use the RemBERT implementation from the Huggingface Transformers Library (Wolf et al., 2020).

C Downstream Hyperparameters

Hyperparameters for downstream tasks are presented in Table 5 and Table 6.

Task	TPU	Batch Size	Learning Rate	Weight Decay	Dropout	Epochs	Warmup Ratio
News Category Classification	False	16	2e-5	0.01	0.1	20	0.10
Wikipedia Section-Title Prediction	True	256	2e-5	0.01	0.1	3	0.10
Named Entity Recognition	True	512	2e-5	0.01	0.1	20	0.10
BBC Hindi News Classification	False	16	2e-5	0.01	0.1	20	0.10
Soham Bengali News Classification	False	16	2e-5	0.01	0.1	8	0.10
INLTK Headlines Classification	False	256	2e-5	0.01	0.1	20	0.10
IITP Movie Review	False	64	5e-5	0.01	0.25	20	0.10
IITP Product Review	False	16	5e-5	0.01	0.5	20	0.10
MIDAS Discourse Mode	False	32	2e-5	0.01	0.5	20	0.10

Table 5: Hyperparameters for ALBERT models

Task	TPU	Batch Size	Learning Rate	Weight Decay	Dropout	Steps	Label Smoothing
News Category Classification	False	16	1e-5	0.1	0.1	2500	0.0
Wikipedia Section-Title Prediction	True	256	8e-6	0.1	0.1	12500	0.0
Named Entity Recognition	False	16	5e-5	0.1	0.1	10000	0.0
BBC Hindi News Classification	False	16	1e-5	0.01	0.1	2500	0.0
Soham Bengali News Classification	False	16	1e-5	0.1	0.1	2500	0.1
INLTK Headlines Classification	False	16	1e-5	0.1	0.1	5000	0.0
IITP Movie Review	False	16	1e-5	0.1	0.1	5000	0.0
IITP Product Review	False	16	1e-5	0.1	0.1	5000	0.0
ACTSA Sentiment Classification	False	16	1e-5	0.1	0.1	5000	0.0
MIDAS Discourse Mode	False	16	8e-6	0.1	0.1	2500	0.1

Table 6: Hyperparameters for RemBERT models

For the ALBERT models batch size was chosen to be the maximum that fits in memory. This was done so that each batch contains approximately the same number of tokens. Otherwise the hyperparameters were chosen following the recommendations of (Mosbach et al., 2021). On the highly skewed IITP Movie Review, IITP Product Review and MIDAS Discourse we found that this default setting resulted in worse performance compared to the independent baselines. So we finetuned the learning rate and classifier dropout on the validation set of these tasks.

For the RemBERT models learning rate, weight decay, dropout, steps and label smoothing were chosen based on grid search with a few values.

D Test Statistics Results

ρ gives us the probability of one group being better than the other group. That is the probability that a random performance sample of the the uni-script model is greater than a random performance sample of the multi-script model. The last test statistic is r which indicates the magnitude of difference between the performance values of the uni-script model (group 1) and the multi-script model (group 2). r shows us how realistically significant the performance differences are between models even if the performance difference is statistically significant. It gives us a value between 0 to 1 and its ranges are: **small effect** ($0 \leq r \leq 0.3$) , **medium effect** ($0.3 < r \leq 0.5$) and **large effect** ($0.5 < r$). We performed MWU on all downstream tasks except CSQA. On CSQA, we only report the δ. The MWU is performed using the SciPy library (Virtanen et al., 2020), and the results are further validated using R (Lüdecke, 2020). These statistic are reported in Table 7 for the IndicGLUE classification tasks and in Table 8 for the public dataset classification tasks.

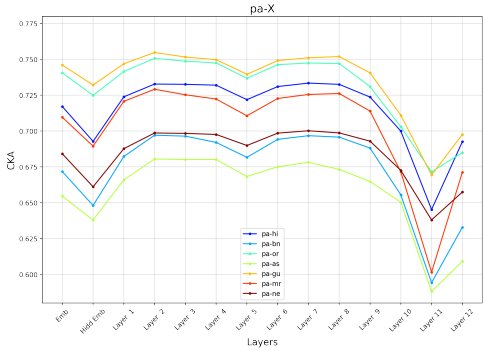
E Cross-lingual Similarity of ALBERT Models on All Language Pairs

Model	pa	hi	bn	or	as	gu	mr	kn	te	ml	ta
Wikipedia Section Title Prediction											
RemBERT$_\rho$	1	1	1	1	1	1	0.91	1	1	0.67	0.99
RemBERT$_r$	0.83	0.83	0.83	0.84	0.83	0.84	0.69	0.83	0.83	0.27	0.81
ALBERT$_\rho$	1	1	1	1	1	1	1	-	-	-	-
ALBERT$_r$	0.83	0.83	0.83	0.83	0.83	0.83	0.83	-	-	-	-
News Category Classification											
RemBERT$_\rho$	1	-	0.27	1	-	0.87	0.46	0.27	0.45	0.94	0.75
RemBERT$_r$	0.85	-	0.39	0.84	-	0.68	0.07	0.39	0.07	0.75	0.41
ALBERT$_\rho$	1	-	0.31	1	-	0.80	0.31	-	-	-	-
ALBERT$_r$	0.86	-	0.32	0.83	-	0.51	0.32	-	-	-	-
Named Entity Recognition											
RemBERT$_\rho$	1.00	0.95	0.99	0.70	0.91	1.00	0.57	1.00	1.00	0.56	0.65
RemBERT$_r$	0.83	0.75	0.81	0.33	0.69	0.83	0.10	0.83	0.83	0.08	0.25
ALBERT$_\rho$	1	1	1	1	1	1	1	-	-	-	-
ALBERT$_r$	0.83	0.83	0.83	0.83	0.83	0.83	0.83	-	-	-	-

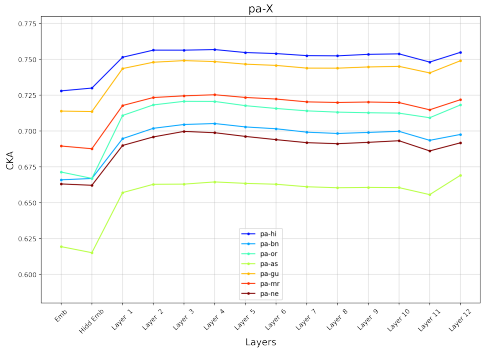
Table 7: Test Statistics on Classification Tasks from IndicGLUE Benchmark

Language	Dataset	RemBERT$_\rho$	RemBERT$_r$	ALBERT$_\rho$	ALBERT$_r$
Article Genre Classification					
hi	BBC News	0.78	0.47	0.87	0.62
bn	Soham News Article Classification	1	0.84	0.87	0.62
gu	INLTK Headlines	1	0.84	0.57	0.12
mr	INLTK Headlines	0.98	0.79	0.36	0.22
ml	INLTK Headlines	0.08	0.70	-	-
ta	INLTK Headlines	0.15	0.59	-	-
Sentiment Analysis					
hi	IITP Product Reviews	0.51	0.01	0.79	0.48
hi	IITP Movie Reviews	0.93	0.72	0.52	0.03
te	ACTSA	0.31	0.30	-	-
Discourse Mode Classification					
hi	MIDAS Discourse	0.79	0.48	0.45	0.07

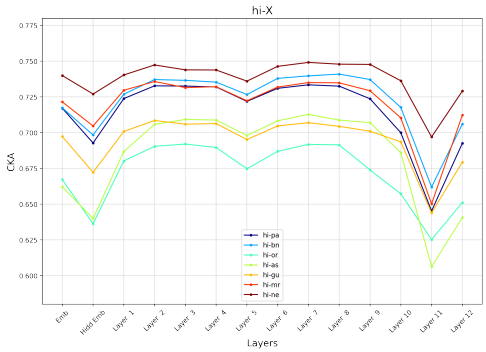
Table 8: Test Statistics on Public Datasets



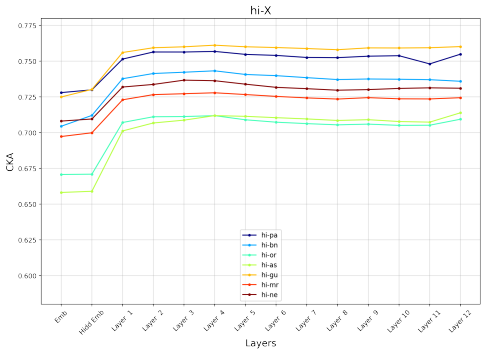
(a) multi-script PA-X



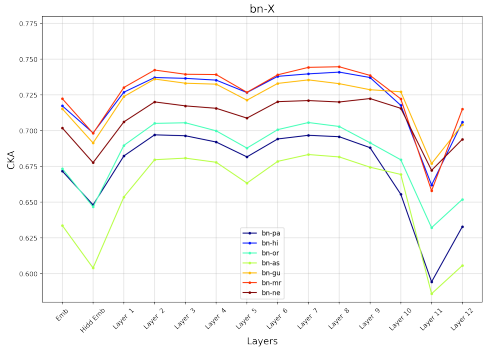
(b) uni-script PA-X



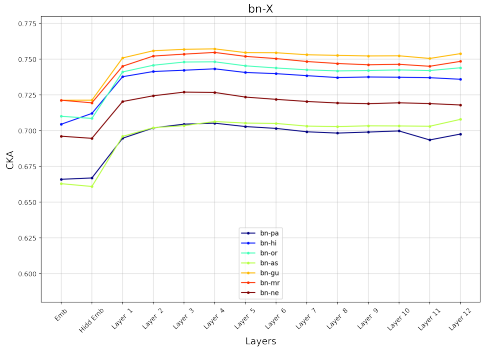
(c) multi-script HI-X



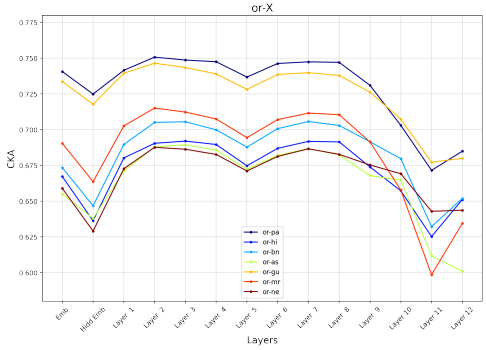
(d) uni-script HI-X



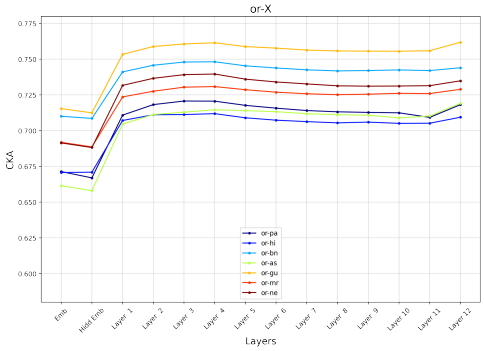
(e) multi-script BN-X



(f) uni-script BN-X

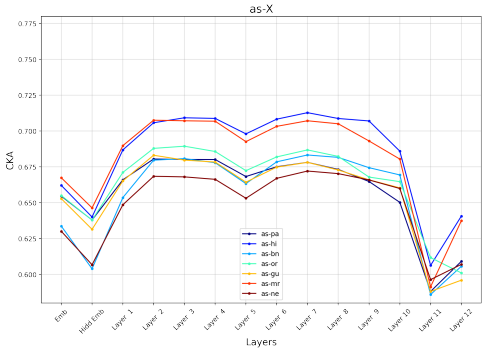


(g) multi-script OR-X

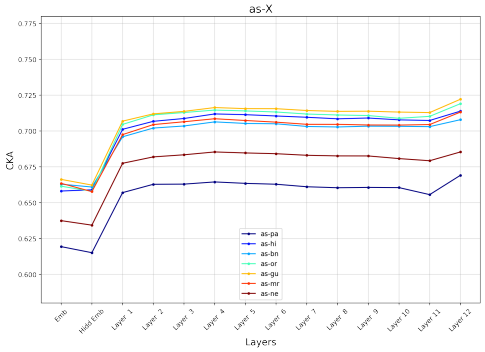


(h) uni-script OR-X

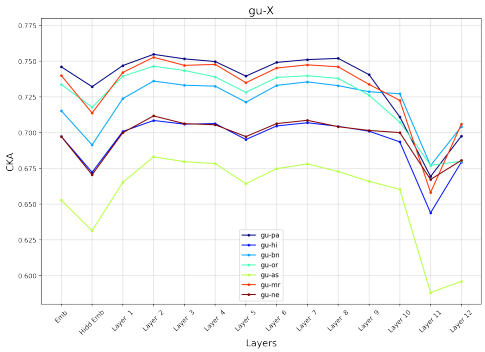
Figure 3: CKA of multi-script and uni-script on all language pairs for pa, hi,bn and or



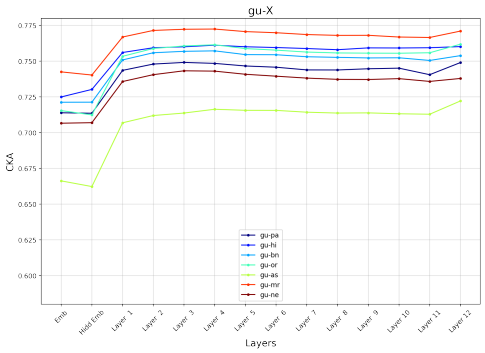
(a) multi-script AS-X



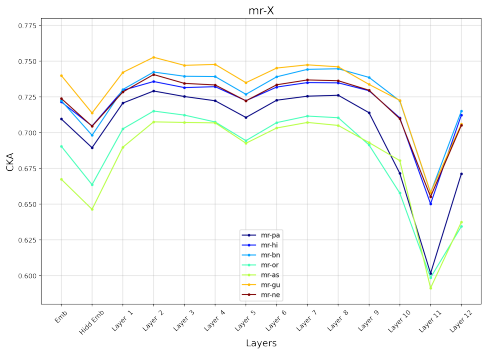
(b) uni-script AS-X



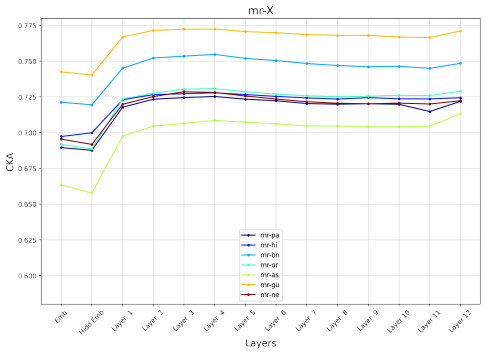
(c) multi-script GU-X



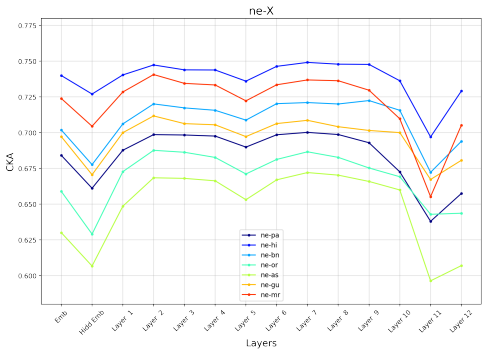
(d) uni-script GU-X



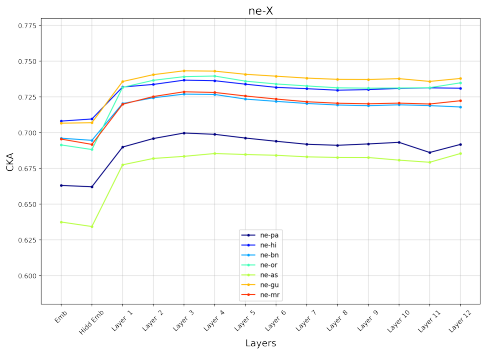
(e) multi-script MR-X



(f) uni-script MR-X



(g) multi-script NE-X



(h) uni-script NE-X

Figure 4: CKA of multi-script and uni-script on all language pairs for AS, GU, MR and NE