IndicXNLI: A Dataset for Studying NLI in Indic Languages

Anonymous ACL submission

Abstract

While Indic NLP has made rapid advances recently in terms of the availability of corpora and pre-trained models, benchmark datasets on standard NLU tasks are limited. To this end, we introduce IndicXNLI, an NLI dataset for 11 Indic languages. It has been created by high-quality machine translation of the original English XNLI dataset and our analysis attests to the quality of IndicXNLI. By finetuning different pre-trained LMs on this IndicXNLI, we analyze various cross-lingual transfer techniques with respect to the impact of the choice of language models, languages, multi-linguality, mix-language input, etc. These experiments provide us with useful insights into the behaviour of pre-trained models for a diverse set of languages. IndicXNLI will be publicly available for research.

1 Introduction

Natural Language Inference (NLI), also known as textual entailment, is a well-studied NLP task (Dagan et al., 2013) where, given a premise and a hypothesis, the model determines whether the premise implies, negates, or is neutral towards the assertions in the hypothesis. In the current era of representation learning-based NLU models, particularly with transformers (Vaswani et al., 2017) and self-supervised language modelling (Devlin et al., 2019; Radford and Narasimhan, 2018), the task is well suited for evaluating the quality of semantic representations generated by Natural Language Understanding (NLU) models (Dagan et al., 2013). Standard English language NLI datasets like MultiNLI (Williams et al., 2018) and SNLI (Bowman et al., 2015) have contributed to the popularity and relevance of the task to evaluating NLU.

Recently, Multi-lingual NLP has gained much attention with the availability of multi-lingual pre-trained language models like mBERT (Devlin et al., 2019), and XLM-R (Conneau et al., 2020) promising cross-lingual transfer and universal models. However, datasets are generally lacking for most languages. Some multi-lingual datasets such as XNLI (Conneau et al., 2018) for NLI, XQUAD (Dumitrescu et al., 2021), MLQA (Lewis et al., 2020) for question answering, PAWS-X for paraphrase identification (Yang et al., 2019) have tried to address this gap. In many practical cases too, training sets are not available for non-English languages, hence cross-lingual zero-shot evaluation benchmarks like XTREME (Hu et al., 2020), XTREME-R (Ruder et al., 2021) and XGLUE (Liang et al., 2020) have been proposed based on these datasets.

The coverage of Indic languages, spoken in by more than 1 billion people in the Indian subcontinent, is low in many of these datasets. Some efforts have been undertaken recently to create benchmark datasets for Indic languages like the IndicGLUE (Kakwani et al., 2020) benchmark. However, NLI datasets are not available for major Indic languages. The only exceptions are the test/validation sets in the XNLI (hi and ur), TaxiXNLI (hi) (K et al., 2021) and MIDAS-NLI (Uppal et al., 2020) datasets . Furthermore, because MIDAS-NLI is based on sentiment data recasting, hypotheses are not linguistically diverse and span limited reasoning.

In this work, we address this gap by introducing IndicXNLI, an NLI dataset for *Indic* languages. IndicXNLI consists of English XNLI data translated into eleven *Indic* languages. We use IndicXNLI to evaluate several *Indic*-specific models (trained only on *Indic* and English languages) such as IndicBERT (Kakwani et al., 2020) and MuRIL (Khanuja et al., 2021), as well as generic (train on several non-*Indic* languages) such as mBERT(cased/uncased) and XLM-RoBERTa. Furthermore, we experimented with several training strategies for each multi-lingual model. Our experimental results answers multiple important questions regarding effective training for *Indic* NLI. In summary, our contributions are as follows:

- We introduce INDICXNLI, a challenging NLI benchmark dataset comprising of NLI data for eleven prominent *Indic* languages from Indo-Aryan branch of Indo-European family and Dravidian family, the two prominent language families in the subcontinent.
- On the INDICXNLI dataset, we investigate several strategies to train multi-lingual classifiers for NLI tasks on INDICXNLI.
- We also explore multi-lingual models cross-lingual NLI transfer performance across all eleven *Indic* languages of INDICXNLI.
- Furthermore, we investigate multi-lingual models performance on EN-INDICXNLI task which contains English premises with corresponding *Indic* hypothesis.

The INDICXNLI dataset, along with associated model scripts, is available at `anonymous_for_submission`.

2 The INDICXNLI dataset

We created INDICXNLI, a NLI corpus for *Indic* languages. INDICXNLI is similar to existing XNLI dataset in shape/form, but focusses on *Indic* language family. INDICXNLI include NLI data for eleven major *Indic* languages that includes Assamese ('as'), Gujarat ('gu'), Kannada ('kn'), Malayalam ('ml'), Marathi ('mr'), Odia ('or'), Punjabi ('pa'), Tamil ('ta'), Telugu ('te'), Hindi ('hi'), and Bengali ('bn'). The next sections details the INDICXNLI construction and its validation.

2.1 INDICXNLI Construction

To create INDICXNLI, we follow the approach of the XNLI dataset and translate the English XNLI dataset (premises and hypothesis) to eleven *Indic*-languages. We use the IndicTrans (Ramesh et al., 2021), a state-of-the-art, publicly available translation model for Indic languages, for machine-translating from English to *Indic* languages. The train (392,702), validation (2,490), and test sets (5,010) of English XNLI were translated from English into each of the eleven *Indic* languages.

IndicTrans is a large Transformer-based sequence to sequence model. It is trained on Samanantar dataset (Ramesh et al., 2021), which is the largest publicly accessible parallel multilingual corpus for these eleven *Indic* languages. IndicTrans outperforms other open-source models based on mBART (Liu et al., 2020) and mT5 (Xue et al., 2021) for *Indic* language translations and is competitive with paid translation models such as Google-Translate[1] or Microsoft-Translate[2] on some benchmarks. Our choice of IndicTrans was motivated by factors of *cost, language coverage and speed*. We have discussed more in detail in Appendix §A.

2.2 INDICXNLI Validation

While translation runs the risk of not preserving the semantic relation between the sentences in the pair, previous work indicates that this is a minimal concern (K et al., 2021). Further, K et al. (2021) provide qualitative analysis to show that classification labels, as well as reasoning categories, are minimally affected for machine-translated NLI datasets given a good quality MT system. Further, we show that the translations generated by IndicTrans are of good quality in two ways (a.) automatic metric BertScore (Zhang* et al., 2020) and , (b.) manual human validation . Given this, we can be confident that most of the classification labels in INDICXNLI are correct. The remainder of this section describes the validation of IndicTrans translation quality.

Automatic Validation Given the absence of *Indic* language XNLI reference data, we use BERTScore similarity between the original English and round-trip translated English sentences for automatic evaluation. The round-trip translated English data is obtained by translating the INDICXNLI test set to English using the IndicTrans model. This evaluation approach estimates the upper bound of the English to *Indic* translation errors, as it approximates the combined error of both English to *Indic* translation, and *Indic* to English translation (Rapp, 2009; Miyabe and Yoshino, 2015; Edunov et al., 2020; Behr, 2017).

We use BERTScore for evaluation because it correlates better with human judgment at the sentence level (Zhang* et al., 2020) compared to BLEU (Papineni et al., 2002). While BLEU computes exact word-level lexical match, BertScore computes a word-level semantic similarity. In Table 1 we compare BERTScore (F1 score) between IndicTrans and Google-Translate round-trip English data. Because Google-Translate does not support Assamese, we do not provide the BERTScore.

We see that the BERTScore for Google Translate and IndicTrans are comparable. Except for

[1] `https://pypi.org/project/googletrans`
[2] `https://github.com/MicrosoftTranslator/`

Malayalam ('ml'), Google Translate looked to be perfect for all languages. We also discovered that BERTScore correlates to resource variability, i.e. better for a high resource than a low resource. High BERTscores validate the quality of IndicTrans translation and, in turn, justify the quality of the INDICXNLI dataset.

Language	hi	te	pa	bn	as	gu
BertScore^{GT}	1.0	1.0	1.0	1.0	NA	1.0
BertScore^{IT}	0.98	0.94	0.94	0.98	0.93	0.94
Human Eval	0.95	0.95	0.94	0.93	0.91	0.90
Language	**ta**	**ml**	**kn**	**mr**	**or**	-
BertScore^{GT}	1.0	0.97	1.0	1.0	1.0	
BertScore^{IT}	0.94	0.94	0.94	0.93	0.93	
Human Eval	0.90	0.85	0.84	0.84	0.83	-

Table 1: BertScore (F1 Score) Befor back-translation with Google-Translate (BertScore^{GT}) and IndicTrans (BertScore^{IT}) translation model. Human evaluation (Human Eval) scores by *Indic* proficient annotators.

Human Validation We followed SemEval-2016 Task-I (Agirre et al., 2016) guidelines for the human validation. Below, we describe the human validation process:

Hiring Experts: We hired eleven annotators who are native speakers in each of the eleven *Indic* languages. These experts annotators are bilingual (English, *Indic*) and proficient in reading/writing for mother-tongue *Indic* and English language.

Diverse Sampling: Since human validation is time-consuming and expensive. We sampled 100 diverse sentences of the test set for validation. We apply the Determinantal Point Process (Kulesza, 2012) over sentence representation for sampling. DPP maximizes coverage volume using a minimal sampled set, therefore guaranteeing diversity in sampling. We first used sentence transformers to convert data to their respective BERT Embeddings, and then use k-DPP (Kulesza and Taskar, 2011) with k = 100 to sample 100 vectors from these embeddings[3]. Using DPP for diverse sampling is a cost-effective method of evaluating translation quality. We have discussed more in detail the scoring guidelines in Appendix §B.

Evaluation: Table 1 shows the final human evaluation scores. Overall, we observe that for all languages, the human scores > 0.83. For high resource languages such as 'hi', 'te', 'pa', 'as', and 'gu' the scores are between 0.90 and 0.95. On the other hand, low resource languages such as 'ta', 'ml', 'kn', 'mr', and 'or' these scores are between 0.83 and 0.90. High human scores reinforce IndicTrans translation quality and indicate excellent INDICXNLI data quality.

3 Experiments and Results

The objective of our experiments is to study how different multi-lingual models, including the one trained specifically for *Indic* languages perform on the INDICXNLI dataset. We first discuss several multi-lingual models explored in our study.

Multi-lingual models. For our experiments, we consider two categories of multi-lingual models, (a) *Indic* specific: these models are specially pre-trained using Mask Language Modeling (MLM) or Translation Language Model (TLM) (Conneau and Lample, 2019) on monolingual / bilingual *Indic* language corpora. These include models such as IndicBERT and MuRIL, and (b) Generic: include massive multi-lingual models pre-trained large number of languages (typically around 100) with MLM such as XLM-RoBERTa and mBERT.

Indic *specific*: These include models such as MuRIL and IndicBERT trained on 17 and 11 *Indic* languages (+English) respectively. MuRIL is pre-trained using Common-Crawl Oscar Corpus (Ortiz Su'arez et al., 2019), PMIndia (Haddow and Kirefu, 2020) on the following languages: *en, hi, bn, ta, ur, ml, te, mr, new, kn, gu, pa, sd, or, as, say, ks*. IndicBERT is pre-trained using *Indic*-Corp (Kakwani et al., 2020) on the following languages: *en, hi, bn, ta, ml, te, Mr, kn, gu, pa, or, as*. Moreover, MuRIL is also pre-trained with TLM objective (with MLM objective) on machine translated data and machine transliterated data.

Generic: These include models such as multi-lingual BERT i.e. mBERT (cased/uncased) and multi-lingual RoBERTa i.e. XLM-RoBERTa which are train on a large number of languages. XLM-RoBERTa also includes pre-training on all eleven *Indic* languages. XLM-RoBERTa is pre-trained using the common crawl monolingual data. mBERT (cased/uncased) includes pre-training on nine of eleven *Indic* languages (Assamese and Odia are not included in pre-training) and uses multi-lingual Wikipedia data for pre-training.

[3] We used the dppy[4] python library for k-DPP.

For all the discussed multi-lingual models, we build NLI classifiers by finetuning the pre-trained models. The classifier takes two sentence as input, i.e. the premise and the hypothesis as input and predicts the inference label. See Appendix §C for model hyper-parameters details.

Training-Evaluation Strategies To train the NLI classifier, we investigate several strategies. These strategies differ on the dataset we used for training and evaluation while keeping the underlying pre-trained multi-lingual models constant. Next, we describe these strategies in detail.

1. ***Indic* Train**: The models are trained and evaluated on INDICXNLI. This is the *translate-train* scenario since the training set is translated from the original English dataset.
2. **English Train**: The models are trained on original English XNLI data and evaluated on INDICXNLI data. This is a *zero-shot evaluation* training scenario.
3. **English Eval**: The model are trained on original English XNLI data, but evaluated on English translation of INDICXNLI data. This is the *translate-test* scenario.
4. **English + *Indic* Train**: This approach combines approaches (1) and (2). The model is first pre-finetuned (Lee et al., 2021; Aghajanyan et al., 2021) on English XNLI data and then finetuned on **individual *Indic* language** of INDICXNLI data.
5. **Train All**: This approach first pre-finetunes the pre-trained model on English XNLI data followed by training on **all the eleven *Indic* languages** jointly.

For all strategies, the development set of INDICXNLI is similar, i.e. in the same language as the evaluation set of INDICXNLI.

3.1 INDICXNLI Results and Analysis

In this section, we discuss the performance (accuracy) of multi-lingual models with varying training strategies for INDICXNLI inference task. We try to answer the following research questions:

1. **RQ1:** How does models perform on INDICXNLI. Are *Indic* languages pre-trained (i.e. *Indic*-specific) model better? (§3.1.1)
2. **RQ2:** Is it desirable to train and evaluate the models on the English translated INDICXNLI data? (§3.1.2)
3. **RQ3:** Can we enhance models performance on INDICXNLI using English XNLI as additional training data? (§3.1.3)
4. **RQ4:** Is the performance of the unified *Indic* model better than the independent language specific *Indic* models? (§3.1.4)

3.1.1 INDICXNLI multi-lingual models. (RQ1)

This correspond to the ***Indic* Train** setting, where model is train and evaluated on each *Indic* languages independently. Table 2 shows the multi-lingual models performance.

Results Analysis. We observe that MuRIL shows the best average performance; this can be attributed to two reasons, the model (a.) is pre-trained on *Indic* languages, (b.) and has more parameters, i.e. deeper architecture with bigger embedding size. On average most models give their best NLI performance on *Hindi* (hi) language set of INDICXNLI. Furthermore, *Odia* (or) language set of INDICXNLI, seem most challenging. On *Odia* (or) larger multi-lingual models such as mBERT (cased/uncased) struggles for good performance. The poor performance of mBERT on Odia can be attributed to its arcane script (Pires et al., 2019). The poor performance can be attributed to the fact at mBERT, which can be attributed to the nature of script of *Odia*. XLM-RoBERTa is at par with MuRIL despite being a generic model. The performance gains are maximum for the *Hindi* (hi) language. This is because, among all these languages, *Hindi* (hi) had the highest proportion of pre-training data on models, resulting in better improvements for the NLI models when trained on *Hindi*. IndicBERT, despite being a smaller model, performs as good as mBERT (cased/uncased). This can be attributed to the *Indic*-specific nature of the IndicBERT model.

3.1.2 How well English XNLI train model perform on INDICXNLI? (RQ2)

Next, we discuss how we can leverage original English XNLI data for model training. We choose English because it is the most prominent language set on which models are (a.) pre-trained using MLM or TLM objective with English corpus, (b.) trained for Multi-Task objective for multiple tasks with English benchmark dataset, (c.) better at cross-lingual transferability with English training (Hu et al., 2020) .

English Train: To test this, we experiment with **English Train** model which is train on the original English XNLI data, and evaluated for *cross lingual transfer* performance on the *Indic* languages in INDICXNLI set. Training over high-resource English language benefit model for effective NLI task-adaptation. Table 2 shows the models performance.

Results Analysis. On average, for all models, the cross-lingual transfer performance is best for *Bengali* (bn) and *Hindi* (hi) language. One possible explanation for this high-performance level is because multi-lingual models are pre-trained on quite large monolingual corpora of these two languages. Here too, MuRIL model performs best across most languages, while cross-lingual transfer on *Hindi* (hi) is best for most models.

When, exclusively using English XNLI data for training, the model's overall performance is worse than *Indic*-specific language is used for training (i.e. ***Indic* Train**), refer §3.1.1 Table 2. We suspect this poor performance is because model fail to understand language-specific features. This proves the requirement of our *Indic* languages specific INDICXNLI data for effective model training. However, the drop in performance was not drastic in comparison with the ***Indic* Train** setting indicating that cross-lingual transfer after fine-tuning on English XNLI data is a strong baseline.

Despite lesser parameters, e.g. IndicBERT outperforms mBERT (cased/uncased), MuRIL outperforms XLM-RoBERTa. As earlier, in comparison to *Indic* pre-train models such as MuRIL and IndicBERT, both XLM-RoBERTa and mBERT (cased/uncased) perform particularly poorly with *Odia* (or) language. From this, we can infer that *Indic*-specific pre-training is beneficial for the cross-lingual transfer task.

English Eval : We further enhance **English Train** cross-lingual transferability, using English translated INDICXNLI evaluation set. To obtain an evaluation set in the English language, we use the IndicTrans translation model. The model performs *Indic* to English translation of the INDICXNLI evaluating sets. This method of *evaluation translation* effectively bridges the linguistic gap due to language variance between the training and the evaluation set. Table 2 shows the multi-lingual model performance.

Results Analysis: We observe that the performance of the multi-lingual model improves when tested on translated English data. This improvement is attributed to the model being trained and assessed on homogeneous resource-rich English language data. Furthermore, the models perform much better on *Odia* (or) language when compared with previous strategies. Despite, substantial gain on *Odia* (or), models still performs best for resource-rich *Hindi* (hi) and *Bengali* (bn) languages. The two reasons for this performance variation across languages are (a.) weaker *Indic*-English translation by IndicTrans for low-resource *Indic* languages, (b.) and, better pre-training (due to larger share in pre-training data) for *Hindi* (hi) and *Bengali* (bn) languages.

XLM-RoBERTa appears to be the best model, which is unexpected given MuRIL's is *Indic*-specific and of similar size (similar number of parameters). This shows that generic models perform better in English evaluation settings as compared to *Indic*-specific models. The fact that the assessment and pre-training language are both English benefits these generic models.

3.1.3 Does Pre-finetuning on English XNLI help multi-lingual models? (RQ3)

Several studies has shown that *Pre-finetuning* approach (Lee et al., 2021; Aghajanyan et al., 2021) i.e. early training a pre-trained model on similar task using augmented data benefits low-resource generalization through effective task-adaptation. We also use the English XNLI data as augmented training data for *"initial fine-tuning"* of model. We use the **English + *Indic* Train** model, which is first trained on English XNLI data followed by training on **individual *Indic* language** of INDICXNLI. We use the same *Indic* language for both training and evaluation for **English + *Indic* Train** model.

Firstly, training on high resource English XNLI data ensure models better adapt for the NLI task. Followed by training on the *Indic* dataset, support the models in acquiring language specific aspects and cross-lingual transfer ability (Xu et al., 2021; Gururangan et al., 2020; Aghajanyan et al., 2021). Thus we effectively combine the **English Train** model (for task adaptation) and ***Indic* Train** model (for cross lingual and language-specific learning) in the **English + *Indic* Train** model setting. Table 2 shows the multi-lingual models performance.

Results Analysis: Overall, we observe that English followed by *Indic* language training tends to en-

Strategy	Model	as	gu	kn	ml	mr	or	pa	ta	te	bn	hi	ModelAvg
***Indic* Train** §3.1.1 RQ1	**XLM-RoBERTa**	**70**	73	**75**	70	**75**	32	71	76	76	**76**	**78**	70
	IndicBERT	67	69	68	60	68	69	**73**	37	62	70	68	65
	mBERT-cased	71	62	69	71	71	35	70	70	69	67	**74**	66
	MuRIL	70	**78**	**75**	**76**	70	**76**	72	**74**	**78**	75	71	**74**
	mBERT-uncased	64	64	63	66	65	35	68	67	67	62	**72**	63
	LanguageAvg	68	69	70	69	70	49	71	65	70	70	**72**	68
English Train §3.1.2 RQ2	**XLM-RoBERTa**	65	**66**	69	**69**	67	**67**	61	71	69	69	**73**	69
	IndicBERT	57	63	53	42	59	57	**66**	41	56	48	63	60
	mBERT-cased	51	57	57	57	54	34	59	**61**	59	57	67	59
	MuRIL	**68**	32	**75**	34	**68**	**67**	**70**	**74**	**71**	**74**	**76**	**72**
	mBERT-uncased	49	55	**64**	59	57	35	58	60	58	62	61	55
	LanguageAvg	58	55	64	52	61	52	63	61	63	62	**68**	63
English Eval §3.1.2 RQ2	**XLM-RoBERTa**	**66**	**72**	70	**68**	**66**	65	**72**	69	**72**	71	**75**	**70**
	IndicBERT	63	66	68	61	65	65	66	63	63	**72**	**72**	66
	mBERT-cased	62	64	67	65	61	60	66	**67**	66	**75**	72	66
	MuRIL	65	33	**71**	67	67	**67**	71	31	71	72	**77**	63
	mBERT-uncased	61	65	61	65	56	66	69	70	67	**76**	74	66
	LanguageAvg	64	60	68	65	63	64	69	60	68	73	**74**	66
English+*Indic* Train §3.1.3 RQ3	**XLM-RoBERTa**	**73**	75	**77**	75	**74**	73	75	75	73	**75**	**79**	76
	IndicBERT	67	72	65	62	59	59	**74**	63	66	69	**74**	70
	mBERT-cased	67	70	69	70	70	39	71	**73**	70	70	71	69
	MuRIL	76	**77**	77	**79**	74	**76**	**77**	**77**	**74**	**75**	**77**	**77**
	mBERT-uncased	64	69	63	73	67	35	68	69	68	72	**74**	69
	LanguageAvg	69	73	70	72	68	56	73	72	70	72	**75**	**72**
Train All §3.1.4 RQ4	**XLM-RoBERTa**	**73**	**77**	**74**	**76**	72	73	**77**	**77**	**76**	**77**	**77**	75
	IndicBERT	63	74	59	51	69	66	**75**	60	67	70	74	66
	mBERT-cased	63	69	69	71	70	33	71	69	70	**74**	72	66
	MuRIL	**73**	76	**74**	**76**	**74**	**78**	**81**	**78**	**76**	**80**	**78**	**77**
	mBERT-uncased	67	70	69	67	67	40	71	73	67	**75**	72	67
	LanguageAvg	68	73	69	68	71	58	75	71	71	**75**	74	70

Table 2: Here, LanguageAvg represents the language wise average score for all models, while ModelAvg average score represents the average score of the model across all languages. Values in **Blue** represents the model wise average best score across languages, while **Red** represents language-wise average best score across models and **Green** represents the values where model-wise and language-wise best score coincide.

hance the performance for all models. As earlier, MuRIL gives the best performance average for most languages, and *Hindi* has the best performance on average across all models. The technique has the best overall accuracy of 72 i.e. aggregated average overall models and languages. The sole downside of this technique is that it has double training time due to both English and *Indic* language training. Moreover, mBERT (uncased/cased) still perform poorly on *Odia* (or) language.

Again, we observe an evident performance benefit to *Indic*-specific models, because of similar reasons as described in the previous section §3.1.1. Moreover, *Indic*-specific models reap benefits of having evaluation data in *Indic* language. We also observe the reduced variance in performance across languages for all models. Despite this, the performance for high resource *Hindi* (hi) and *Bengali* (bn) languages remains the best.

3.1.4 Unified INDICXNLI multi-lingual inference model. (RQ4)

Until recently, we had been creating independent inference models for each *Indic* language through various settings. However, prior work on translation has demonstrated that multi-lingual models trained together on multiple closely related languages always perform better than individual bilingual models (Ramesh et al., 2021). On similar lines, we increase the languages exposure for NLI models by training the model on **all the eleven *Indic* languages together** i.e. **Train All** setting. In **Train-all** we create a unified multi-lingual model by first training on English XNLI followed by training the same model on **all the eleven *Indic* languages** i.e. complete *Indic* language family of INDICXNLI.

This **Train All** techniques has multiple benefits, as follows (a.) single unified model work across all *Indic* languages, instead of language-specific several individuals models. (b.) Overall training time is also drastically reduced, compare to

English + *Indic* Train model the **Train All** model is 2× faster to train. (c.) since the same model has trained for all *Indic* languages at once, and the model performs consistently across all languages. For individual models, the amount of pre-training data available in each language can substantially impact their performance. (d.) and model exploit inter-language similarities for better cross-lingual transfer capacity. Table 2 show performance of multi-lingual models.

Results Analysis. Overall we observe that single unified model **Train All** perform much better than individual models i.e. **English + *Indic* Train**, refer to §3.1.4. This lends credence to the argument that unified models developed for closely related languages outperform individual models developed for each language (Tan et al., 2019). The **Train All** method may alternatively be viewed as an extension of English XNLI augmentation, now with remaining INDICXNLI *Indic* languages as additional augmentation data. MuRIL performs best for all languages across models on average. As earlier *Hindi* and *Bengali* has better performance as compared to other *'Indic'* languages.

3.2 INDICXNLI Cross-Lingual Transfer

In this section, we try to answer the following research question.

RQ5: Can language specific model (§3.1.1) transfer performance across *Indic* languages?

In response to RQ5, we evaluate language specific model (§3.1.1) on their Cross-lingual transfer ability across *Indic* languages i.e. evaluating performance of model train on "X" *Indic* language on "Y" *Indic* language. We trained the model with the ***Indic*** Train setting. However, we evaluated each *Indic* language model performance across all ***Indic*** languages. Table 3 present the average evaluation score of all *Indic* language when train on the mentioned column language of INDICXNLI. For detailed model-wise cross-lingual train-test language results, refer to Appendix §D.

Results Analysis: As earlier, we observe that the models tend to favour high resource languages *Hindi* and *Bengali* training for better cross-lingual transfer. Because of the higher amount of monolingual corpus, more frequent pre-training for these languages may be one explanation for improved performance (Conneau et al., 2020). One can think of this as replacing English training §3.1.2 with *Hindi* and *Bengali* training. Furthermore, model train on non-*Hindi* and non-*Bengali* when evaluated for all *Indic* languages perform best for the *Hindi* and *Bengali* language. Except for MuRIL and IndicBERT, which gain from *Indic*-specific pre-training, *Odia* is a difficult language for all other models. MuRIL performs the best amongst all models. We observe a strong correlation between *Indic*-specific pre-training and model performance. We also observe that larger model size and *Indic*-specificity benefit model performance.

From detailed results in Appendix §D, we observe that models have relatively low diagonal correlation, i.e. models may not necessarily perform best on evaluation on the training language. This also demonstrates that selecting the appropriate language for cross-lingual transfer can significantly boost the odds of obtaining a better overall model.

3.3 EN-INDICXNLI Results and Analysis

In this section, we try to answer the following research question. We refer to EN-INDICXNLI as NLI task where the premise is in English and Hypothesis is in *Indic* language.

RQ6: How does INDICXNLI models (§3.1.3 and §3.1.4) perform on EN-INDICXNLI?

In response to RQ6, we analysed the performance of multi-lingual models when premise is in English and hypothesis is in *Indic* language. This task assesses the model's ability to perform abreast (English-*Indic*) **intra-input** cross-lingual reasoning. Therefore, we create EN-INDICXNLI dataset which contain English *premises* from XNLI and corresponding *Indic* hypothesis from INDICXNLI. To asses this task, we train model on EN-INDICXNLI train set using **English + *Indic* Train** (§3.1.3) and **Train All** (§3.1.4) strategies except with English premises. During inference we evaluate on similar setting i.e. EN-INDICXNLI evaluation set. Table 4 shows performance of **English + *Indic* Train** and **Train All** models on EN-INDICXNLI.

Results Analysis. We observed a performance loss except for XLM-RoBERTa when the model is evaluated on EN-INDICXNLI inference task. The inference models struggle to correlate and reason together on two different languages (English, *In-*

Strategy	Model	as	gu	kn	ml	mr	or	pa	ta	te	bn	hi	ModelAvg
Indic Train	**XLM-RoBERTa**	66	70	33	34	70	35	68	70	70	71	**72**	60
	IndicBERT	59	**60**	59	54	**60**	**60**	**60**	56	59	58	**60**	59
	mBERT cased	57	59	**60**	59	58	33	59	**60**	59	**60**	**60**	57
	mBERT uncased	59	59	60	59	58	33	59	60	60	59	**61**	57
	MuRIL	**75**	**73**	**75**	**76**	**71**	33	**75**	**76**	**73**	**75**	**73**	**70**
	LanguageAvg	63	64	57	56	63	39	64	64	64	**65**	**65**	60

Table 3: Summary of *Indic* Cross-Lingual Transfer model performance (refer §3.2 RQ5). Every row represent the average evaluation score of all *Indic* language when train on the mentioned column language of INDICXNLI. For detail results on *Indic* Cross-lingual Transfer refer to Appendix §D. Here, ModelAvg, LanguageAvg, and Color Code mean same as in table 2.

Strategy	Model	as	gu	kn	ml	mr	or	pa	ta	te	bn	hi	ModelAvg
English+*Indic* Train	**XLM-RoBERTa**	**74**	**72**	**75**	**74**	**77**	**72**	70	**72**	**72**	**79**	**76**	**74**
	IndicBERT	70	68	63	65	69	68	**71**	64	64	69	69	67
	mBERT-cased	51	56	59	50	62	31	**63**	57	60	61	**63**	56
	MuRIL	71	70	**73**	69	71	39	71	71	69	72	69	67
	mBERT-uncased	60	57	61	61	59	56	36	59	69	**74**	71	60
	LanguageAvg	65	65	66	64	68	53	62	65	67	**71**	70	65
Train All	**XLM-RoBERTa**	**57**	**59**	**58**	**62**	**61**	53	57	59	**61**	**63**	**63**	**59**
	IndicBERT	49	53	46	37	52	51	59	39	51	**57**	50	50
	mBERT-cased	39	39	43	38	**43**	33	40	42	41	40	42	40
	MuRIL	51	52	58	56	53	**55**	**58**	**65**	55	62	54	56
	mBERT-uncased	40	42	49	46	48	40	46	45	45	**48**	44	44
	Language-Avg	47	49	51	48	51	45	52	50	51	**54**	51	50

Table 4: EN-INDICXNLI model performance (refer §3.3 RQ6) with English + *Indic* train and Train All setting. Here, ModelAvg, LanguageAvg, and Color Code mean same as in table 2.

dic) sentences. Contrary to earlier observation, a generic model such as XLM-RoBERTa outperforms the *Indic* specific models. However, IndicBERT and MuRIL perform better than mBERT. *Bengali* perform best for both the training strategies. We also observe the benefit of English data augmentation **English +** ***Indic*** **Train** model, rather than all language augmentation **Train All** model.

4 Related Work

Recently many *Indic*-specific resources are developed such as IndicNLPSuite (Kakwani et al., 2020), which include Indic specific (a.) word embeddings: IndicFT, (b.) transformer models: IndicBERT, (c.) monolingual corpora: IndicCorp, (d.) and, evaluation benchmark: IndicGLUE

Furthermore, *Indic*-specific pre-processing libraries such as iNLTK (Arora, 2020) and Indic-nlp-library (Kunchukuttan, 2020), other Indic monolingual corpora: Common Crawl Oscar Corpus (Wenzek et al., 2020; Ortiz Suárez et al., 2020), multilingual parallel corpora: PMIndia (Haddow and Kirefu, 2020) and Samantar (Ramesh et al., 2021), large transformer model MuRIL (Khanuja et al., 2021) and language specific Indic-Transformers (Jain et al., 2020) also exists.

5 Conclusion

Dataset. With INDICXNLI we extend the XNLI dataset for *Indic* languages family. Furthermore, INDICXNLI can also be evaluated for cross-lingual transfer task. We also introduce the challenge language mixed EN-INDICXNLI inference task.

Benchmarks. We analyse how various multilingual models both *Indic*-specific and *Indic*-generic perform on INDICXNLI under various training regime. We study the effects of using English XNLI as training and pre-finetuning data. We also analyse how models perform on Indic Cross-Lingual Transfer tasks. Moreover, evaluation on EN-INDICXNLI further evaluate models intra-input cross-lingual reasoning ability.

Future Work. We aim to integrate INDICXNLI and explore baseline in IndicGLUE benchmark of IndicNLPSuite (Kakwani et al., 2020) library. We also intend to enhance INDICXNLI by enhancing human interaction and trying more advanced translation techniques. It would be interesting to try bigger models such as XLM-RoBERTa$_{Large}$ and MuRIL$_{Large}$ on INDICXNLI. Another direction could be assessing models performance on INDIC-INDICXNLI task, where premises and hypothesis are in two distinct *Indic* languages.

References

Armen Aghajanyan, Anchit Gupta, Akshat Shrivastava, Xilun Chen, Luke Zettlemoyer, and Sonal Gupta. 2021. Muppet: Massive multi-task representations with pre-finetuning.

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. SemEval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 497–511, San Diego, California. Association for Computational Linguistics.

Gaurav Arora. 2020. iNLTK: Natural language toolkit for indic languages. In *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*, pages 66–71, Online. Association for Computational Linguistics.

Dorothée Behr. 2017. Assessing the use of back translation: the shortcomings of back translation as a quality testing method. *International Journal of Social Research Methodology*, 20(6):573–584.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. *Advances in Neural Information Processing Systems*, 32:7059–7069.

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

Raj Dabre, Himani Shrotriya, Anoop Kunchukuttan, Ratish Puduppully, Mitesh M. Khapra, and Pratyush Kumar. 2021. Indicbart: A pre-trained model for natural language generation of indic languages.

Ido Dagan, Dan Roth, Mark Sammons, and Fabio Massimo Zanzotto. 2013. Recognizing textual entailment: Models and applications. *Synthesis Lectures on Human Language Technologies*, 6(4):1–220.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Stefan Daniel Dumitrescu, Petru Rebeja, Beata Lorincz, Mihaela Gaman, Andrei Avram, Mihai Ilie, Andrei Pruteanu, Adriana Stan, Lorena Rosia, Cristina Iacobescu, Luciana Morogan, George Dima, Gabriel Marchidan, Traian Rebedea, Madalina Chitez, Dani Yogatama, Sebastian Ruder, Radu Tudor Ionescu, Razvan Pascanu, and Viorica Patraucean. 2021. Liro: Benchmark and leaderboard for romanian language tasks. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Sergey Edunov, Myle Ott, Marc'Aurelio Ranzato, and Michael Auli. 2020. On the evaluation of machine translation systems trained with back-translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2836–2846, Online. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Barry Haddow and Faheem Kirefu. 2020. Pmindia – a collection of parallel corpora of languages of india.

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. *CoRR*, abs/2003.11080.

Kushal Jain, Adwait Deshpande, Kumar Shridhar, Felix Laumann, and Ayushman Dash. 2020. Indic-transformers: An analysis of transformer language models for indian languages.

Karthikeyan K, Aalok Sathe, Somak Aditya, and Monojit Choudhury. 2021. Analyzing the effects of reasoning types on cross-lingual transfer performance.

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.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. Muril: Multilingual representations for indian languages.

Alex Kulesza. 2012. Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2-3):123–286.

Alex Kulesza and Ben Taskar. 2011. K-dpps: Fixed-size determinantal point processes. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 1193–1200, Madison, WI, USA. Omnipress.

Anoop Kunchukuttan. 2020. The Indic-NLP Library. `https://github.com/anoopkunchukuttan/indic_nlp_library/blob/master/docs/indicnlp.pdf`.

Hung-yi Lee, Ngoc Thang Vu, and Shang-Wen Li. 2021. Meta learning and its applications to natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Tutorial Abstracts*, pages 15–20, Online. Association for Computational Linguistics.

Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2020. MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330, Online. Association for Computational Linguistics.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. 2020. XGLUE: A new benchmark datasetfor cross-lingual pre-training, understanding and generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6008–6018, Online. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Mai Miyabe and Takashi Yoshino. 2015. Evaluation of the validity of back-translation as a method of assessing the accuracy of machine translation. In *2015 International Conference on Culture and Computing (Culture Computing)*, pages 145–150.

Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2020. A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1703–1714, 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. 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.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. 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.

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pre-training.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, Navneet Kumar, Aswin Pradeep, Kumar Deepak, Vivek Raghavan, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh Shantadevi Khapra. 2021. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages.

Reinhard Rapp. 2009. The back-translation score: Automatic mt evaluation at the sentence level without reference translations. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, ACLShort '09, page 133–136, USA. Association for Computational Linguistics.

Sebastian Ruder, Noah Constant, Jan Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Dan Garrette, Graham Neubig, and Melvin Johnson. 2021. XTREME-R: Towards more challenging and nuanced multilingual evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10215–10245, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xu Tan, Jiale Chen, Di He, Yingce Xia, Tao Qin, and Tie-Yan Liu. 2019. Multilingual neural machine translation with language clustering. 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 963–973.

Shagun Uppal, Vivek Gupta, Avinash Swaminathan, Haimin Zhang, Debanjan Mahata, Rakesh Gosangi, Rajiv Ratn Shah, and Amanda Stent. 2020. Two-step classification using recasted data for low resource settings. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 706–719, Suzhou, China. Association for Computational Linguistics.

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.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Haoran Xu, Seth Ebner, Mahsa Yarmohammadi, Aaron Steven White, Benjamin Van Durme, and Kenton Murray. 2021. Gradual fine-tuning for low-resource domain adaptation. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 214–221, Kyiv, Ukraine. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. 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 3687–3692, Hong Kong, China. Association for Computational Linguistics.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Further Discussions

Why *Indic* languages? Indic languages are spoken by more than a billion people in the Indian subcontinent. With the introduction of IndicNLPSuite (Kakwani et al., 2020) by AI4Bharat there has been has an increased interest and effort towards the research for *Indic* languages model. Recently, IndicBERT, MuRIL (Khanuja et al., 2021) based on BERT (Devlin et al., 2019) were introduced for the *Indic* languages. Furthermore, generation model IndicTrans (Ramesh et al., 2021) and IndicBART (Dabre et al., 2021) based on seq2seq architecture was also published recently. These model use the *Indic* enrich monolingual corpora: Common Crawl, Oscar and IndicCorp and parallel corpora: Samantar and PMIndia (Haddow and Kirefu, 2020) on *Indic* languages for training.

Despite significant progress through large transformer-based *Indic* language models in addition to existing multilingual models e.g. mBERT (Devlin et al., 2019) , XLM-RoBERTa (Conneau et al., 2020), and mBART (seq2seq) (Liu et al., 2020) there is currently a paucity of benchmark data-sets for evaluating these huge language models in the *Indic* language research field. Such benchmark dataset is necessary for studying the linguistic features of Indic languages and how well they are perceived by different multilingual models. Recently, IndicGLUE (Kakwani et al., 2020) was introduced to handle this scarcity. The scope of this benchmark, however, is confined to only few tasks and datasets.

Why Multilingual NLI? Natural Language Inference (NLI) is a task where we are given two sentences, premise and hypothesis and the model has to predict if the premise entails or negates the sentence or does neither. NLI is a classical approach for evaluating the reasoning ability of NLP models. Recently, XNLI (Conneau et al., 2018) a dataset sampled from MultiNLI dataset was created with an intent to evaluate the cross-lingual Multilingual models for several languages. However, this dataset covers only *'Hindi'* in *Indic* languages family. *'Hindi'* although being a prominent language in the Indian subcontinent, is not the native language of many Indians and differs morphologically from languages such as *'Tamil'*, *'Malayali*, and *'Telugu'*, which we considered for this study.

Why INDICXNLI task? This research provides an excellent chance to investigate the efficacy of various Multilingual models on *Indic* languages that are rarely evaluated or explored before. Some of these *Indic* languages such as *'Assamese'* and *'Odia'* serve as unseen (zero-shot) evaluation for models such as mBERT (Pires et al., 2019), i.e. not pre-trained on *'Assamese'*. While other models, such as XLM-RoBERTa, IndicBERT and MuRIL covers all our languages but in widely varying proportions in their training data. Our work investigate the correlation effect of cross-lingual training for English on these rare *Indic* languages, which are not explore by prior studies. Furthermore, we also investigate the cross-lingual transfer effect across *Indic* languages, also not explored before. We explore the impact of Multilingual training, english-data augmentation, unified Indic model performance, cross-lingual transfer of closely related *Indic* family and English-*Indic* NLI through our work. All the above mention topics are not explore for *Indic* language before.

Why IndicTrans for Translation? We use the IndicTrans as a translation model for converting English XNLI to INDICXNLI because of the following reasons.

- **Open-Source:** IndicTrans is open-source to public for non-commercial usage without additional fees, while Google-Translate and Microsoft-Translate require paid subscription.
- **Light Weight:** IndicTrans is the fastest and the lightest amongst mBART and mT5 on single-core GPU machines. Google-Translate and Microsoft-Translate are also relatively slower due to repeated network-intensive API calls.
- ***indic* Coverage:** Seq2Seq models like mBART and mT5 are not designed for all languages in the indic family. mBART supports eight (excludes kn,or,pa,as) while mT5 supports nine languages (excludes or,as) out of eleven indic languages. Google-Translate supports ten out of eleven *indic* languages (excludes Assamese). Microsoft Translate supports all the eleven *indic* languages.

B Human Validation Scoring Details

Finally, we then provide English and *indic* language INDICXNLI (IndicTrans translated) sentence to the recruited native speaker of that *indic* language for validation. Before the annotation work, each expert was given a full explanation of the guidelines that needed to be followed. The validation instructions (mturk template and detailed examples) are taken from the Semeval-2016 Task-I. The native speaker access the sentence pairs assign an integer score between 0 and 5, as follows:

- **0**: The two sentences are completely dissimilar.
- **1**: The two sentences are not equivalent, but are on the same topic.
- **2**: The two sentences are not equivalent, but share some details.
- **3**: The two sentences are roughly equivalent, but some important information differs/missing.
- **4**: The two sentences are mostly equivalent, but some unimportant details differ.
- **5**: The two sentences are completely equivalent, as they mean the same thing.

The score depicts the goodness of translated sentence in terms of semantics, i.e. same meaning as original English sentence[5]. Scores are then normalized to a probability range (between 0 and 1). The final validation score for each language is determined as the average of all 100 instances' scores.

C Hyper Parameters Details

All the models were trained on google collaboratory [6] on TPU-v2 with 8 cores. The code was built in the PyTorch-lightning framework. We used accuracy as mentioned in the original XNLI paper (Conneau et al., 2018) as our metric of choice. The training was run with an early stopping callback with the patience of 3 and validation interval of 0.5 epochs. We used AdamW as our optimizer of choice. (Loshchilov and Hutter, 2019).

D Indic Cross-lingual Transfer

This section is the extension of the §3.2. Table 6, 7, 8, 9, 10 are the cross-lingual transfer results of XLM-RoBERTa, IndicBERT, mBERT-cased, mBERT-uncased and MuRIL respectively. The rows of the table consist of the languages on which

[5] For NLI task, same syntax, i.e. grammar (e.g. Tense) lesser important than same Semantic, i.e. meaning preservation.

[6] https://colab.research.google.com/

Hyper Parameter	XLM-RoBERTa	IndicBERT	MuRIL-cased	mBERT-cased	mBERT-uncased
Learning Rate	2e-5	2e-5	2e-5	2e-5	2e-5
Batch Size	64	128	64	128	128
Weight Decay	0.01	0.01	0.01	0.01	0.01
Max Seq Length	128	128	128	128	128
Model Size	278M	33.7M	237M	177M	167M
Warmup Steps	1500	1500	1500	1500	1500

Table 5: Model Hyper-Parameters and Size (size is described by number of parameters in millions)

the model is trained, while the columns represent the evaluation languages. E.g., in table 7 the first row represents that the model is trained on *"Assamese"* and then tested on all the languages in the column. The values in the row are the accuracy scores of the model when trained on the language in its leftmost column and tested on the language in its top-most row column.

For **XLM-RoBERTa**, the model perform best for the *"Bengali"* language. The model gives the best performance average across all other languages if trained on *"Bengali"*. A model trained in other languages, on average, also performs best for *"Bengali"* language. XLM-RoBERTa also struggles to correlate with *"Kannada"*, *"Odia"*, and *'Malayalam*, thus performs poorly on average if trained for them. At the same time, all models have poor cross-lingual ability transferability for the *"Assamese"* language. Furthermore, XLM-RoBERTa seems to perform better when trained and evaluated for higher resource languages such as *"Bengali"* and *"Hindi"*.

For **IndicBERT**, the overall score is comparable to XLM-RoBERTa despite it being a significantly smaller model. On average, across languages, the cross-lingual transferability for models trained on varying *indic* languages were consistently similar (between 0.5 - 0.6). However, the evaluation performance for cross-lingual models evaluated on *"Malayalam"* were poor for all *indic* trained models. For model trained on some languages, *"Kannada"*, *"Malayalam"* and *"Punjabi"*, the best performance was across diagonal, i.e. indicating the model performs best on the trained language. This trend was, however, surprisingly not accurate in other *indic* languages, indicating remarkable cross-lingual transferability of the IndicBERT model.

For **mBERT-cased**, the model performs worse for **"Odia"** on avergae for both when evaluated and train on. However, all models performs very consistently for other *indic* languages. Model trained on *Kannada*, *Punjabi*, *Tamil*, *Hindi*, and *Bengali* perform best on average across languages. Here too, the best cross-lingual transfer ability was shown for *Bengali* language. mBERT-cased also for some languages have best performance across diagonal, i.e. the model performs the best on the language it is trained on, these languages include *"Assamese"*, *"Gujurati"*, *"Malayalam"*, *"Punjabi"* and *"Telugu"*.

For **mBERT-uncased**, the model correlate poorly for *"Odia"* language, however, shows similar results as mBERT-cased for all other languages. Model trained on *"Kannada"* and *"Bengali"* perform best on average across languages. Here too, the best cross-lingual transferability was shown for *"Bengali"* language. mBERT-uncased also for some languages have best performance across diagonal, i.e. the model performs the best on the language it is trained on, these languages include *"Assamese"*, *"Gujurati"*, *"Malayalam"*, *"Kannada"* and *"Marathi"*.

MuRIL has the best overall cross-lingual transferability amongst all the models. **MuRIL** only fails to generalize well when trained for *"Odia"* language. However, model train on other *indic* language when evaluated on *"Odia"* performs well. Model trained on *Marathi* and *"Marathi"* perform best on average across languages. The best cross-lingual transferability was shown for *"Bengali"* and *"Hindi"* language. Muril shows diagonal correlation in performance with languages such as *"Marathi"*, *"Odia"* and *"Telugu"*. Overall, MuRIL has better cross-lingual transferability across all languages compared to other models. It also reflects less performance bias for languages such as *"Bengali"* and *"Hindi"*, as compared to XLM-RoBERTa.

XLM-RoBERTa	as	gu	kn	ml	mr	or	pa	ta	te	bn	hi	Train Avg
as	64	67	66	67	63	63	**68**	**68**	64	66	65	66
gu	65	72	69	69	68	71	70	71	65	**74**	**74**	70
kn	33	31	**35**	**35**	31	34	32	31	32	33	32	33
ml	**35**	33	33	34	31	34	34	31	33	34	34	33
mr	66	74	70	72	72	68	70	69	65	**75**	73	71
or	35	33	32	36	35	34	34	**36**	34	**36**	**36**	35
pa	65	69	70	67	67	67	70	66	67	**73**	66	68
ta	64	67	69	72	71	68	71	70	70	**73**	70	70
te	61	70	71	70	70	71	68	68	**75**	**75**	72	71
bn	67	72	73	73	72	**74**	**74**	70	70	73	71	**72**
hi	66	70	69	72	69	68	71	71	71	**76**	73	71
Test Avg	56	60	60	61	59	59	60	59	59	**63**	61	60

Table 6: Indic Cross-lingual transfer XLM-RoBERTa

IndicBERT	as	gu	kn	ml	mr	or	pa	ta	te	bn	hi	Train Avg
as	65	63	54	46	61	60	66	48	57	**67**	60	58
gu	61	67	54	41	65	64	**70**	46	62	**70**	62	**60**
kn	58	64	**68**	48	59	59	65	46	59	63	63	59
ml	55	52	54	**60**	53	53	52	52	57	52	52	54
mr	62	65	54	48	61	61	67	52	60	**68**	63	**60**
or	61	66	57	49	61	66	65	48	60	**68**	64	**60**
pa	61	67	55	47	60	62	**74**	41	60	70	62	**60**
ta	55	**60**	53	49	56	54	58	59	55	58	55	56
te	61	63	53	45	59	63	**70**	46	63	68	58	59
bn	62	66	55	48	62	62	66	47	60	**68**	**68**	**60**
hi	58	63	53	49	61	61	66	43	57	**71**	61	59
Test Avg	60	63	55	48	60	60	65	48	59	**66**	61	59

Table 7: Indic Cross-lingual transfer IndicBERT

mBERT-cased	as	gu	kn	ml	mr	or	pa	ta	te	bn	hi	Train Avg
as	**69**	59	61	53	57	36	61	57	52	59	64	56
gu	48	**70**	64	55	60	32	64	64	60	67	65	60
kn	49	62	68	64	60	35	65	64	59	**69**	62	**61**
ml	51	0.60	60	**71**	60	30	61	65	62	66	62	60
mr	45	61	63	56	69	35	64	56	57	**69**	66	60
or	34	33	29	32	36	35	34	35	33	33	34	33
pa	47	65	59	59	62	35	**70**	63	61	68	64	**61**
ta	48	64	**67**	63	60	32	65	66	63	69	62	**61**
te	51	59	63	63	60	32	61	64	**67**	66	62	60
bn	51	64	65	62	62	32	65	60	62	69	**67**	**61**
hi	50	66	65	61	62	30	65	63	61	**71**	63	**61**
Test Avg	49	60	60	58	59	33	61	60	58	**64**	61	58

Table 8: Indic Cross-lingual transfer mBERT-cased

mBERT-uncased	as	gu	kn	ml	mr	or	pa	ta	te	bn	hi	Train Avg
as	**68**	61	56	59	62	32	66	62	55	65	63	58
gu	55	**68**	61	60	60	32	63	63	59	64	67	60
kn	47	62	**72**	65	62	32	66	65	63	65	64	**62**
ml	49	61	59	**67**	56	36	63	65	58	66	63	59
mr	49	60	60	58	**68**	31	62	59	61	69	62	59
or	34	33	29	32	**36**	35	34	35	33	33	34	33
pa	51	62	60	60	62	34	68	61	62	**67**	63	60
ta	53	63	63	64	63	35	65	68	62	**67**	61	61
te	53	61	63	62	62	35	64	62	64	**65**	**65**	60
bn	54	63	61	64	63	34	66	64	62	70	**71**	**62**
hi	47	64	58	61	63	35	64	60	62	**74**	67	61
Test Avg	51	60	58	59	60	34	62	60	58	**64**	62	58

Table 9: Indic Cross-lingual transfer mBERT-uncased

MuRIL	as	gu	kn	ml	mr	or	pa	ta	te	bn	hi	Train Avg
as	73	**78**	75	74	74	73	75	75	75	76	77	75
gu	72	75	75	74	73	72	70	72	71	**76**	75	73
kn	72	75	76	76	73	73	74	75	76	**77**	**77**	75
ml	75	75	73	77	72	78	76	**79**	75	77	76	**76**
mr	69	70	72	71	**73**	68	76	70	69	73	74	72
or	33	36	35	30	32	**35**	30	30	33	32	36	33
pa	73	75	**76**	74	74	76	79	71	74	75	75	75
ta	74	76	76	77	75	72	74	77	76	**80**	78	**76**
te	70	72	74	71	73	70	**77**	74	**77**	**77**	75	74
bn	68	**76**	73	73	71	72	73	74	74	74	**76**	74
hi	73	76	73	75	74	73	76	74	74	75	**76**	75
Test Avg	68	71	71	70	69	69	71	70	70	**72**	**72**	71

Table 10: Indic Cross-lingual transfer MuRIL