Natural Language Processing (2024), 1–18
doi:10.1017/nlp.2024.13

CAMBRIDGE UNIVERSITY PRESS

ARTICLE

Does learning from language family help? A case study on a low-resource question-answering task

Hariom A. Pandya and Brijesh S. Bhatt

Dharmsinh Desai University, Nadiad, Gujarat, India
Corresponding author: Hariom A. Pandya; Email: pandya.hariom@gmail.com

(Received 27 November 2022; revised 26 August 2023; accepted 16 November 2023)

Special Issue on '**Natural Language Processing Applications for Low-Resource Languages**'

Abstract

Multilingual pre-trained models make it possible to develop natural language processing (NLP) applications for low-resource languages (LRLs) using the model of resource-rich languages (RRLs). However, the structural characteristics of the target languages can impact task-specific learning. In this paper, we investigate the influence of structural diversities of languages on the system's overall performance. Specifically, we propose a customized approach to leverage task-specific data of low-resource language families via transfer learning from RRL. Our findings are based on question-answering tasks using the XLM-R, mBERT, and IndicBERT transformer models and Indic languages (Hindi, Bengali, and Telugu). On the XQuAD-Hindi dataset, the few-shot learning using Bengali improves the benchmark mBERT (F1/EM) score by +(10.86/7.87) and XLM-R score by +(3.84/4.42). Few-shot learning using Telugu has also improved the mBERT score by +(10.42/7.36) and +(3.04/2.72) for XLM-R. In addition, our model has demonstrated benchmark-compatible performance in a zero-shot setup with single-epoch task learning. This approach can be adapted for other NLP tasks for LRLs.

Keywords: Question-answering; multilinguality; language family; low-resource learning

1. Introduction

Significant development of natural language processing (NLP) applications has been realized due to the dramatic increase in the availability of pre-trained models. Many popular models such as BERT (Devlin *et al.* 2019), RoBERTa (Liu *et al.* 2019), GPT-3 (Brown *et al.* 2020), and T5 (Raffel *et al.* 2020) have shown promising results in solving important NLP problems like question-answering (QA), summarization, machine translation, sentiment analysis, etc. (Liu, Chen, and Xu 2022; Ofoghi, Mahdiloo, and Yearwood 2022; Suleiman and Awajan 2022; Yadav *et al.* 2022; Zhu 2021b). The availability of such pre-trained models has reduced the requirement for task-specific supervised data, expensive hardware, and large training time. As a result, fine-tuning on a small supervised set is enough for achieving the benchmark results (Zhu 2021a; Zhang *et al.* 2021). The resource-rich languages (RRLs) like English, German, French, etc. are benefited from pre-trained models even in the zero-shot setup (Kaur, Pannu, and Malhi 2021).

On the contrary, the low-resource languages (LRLs) are struggling, with the lack of supervised task-specific data being its bottleneck (Pandya and Bhatt 2021). Figure 1 shows the comparison between a number of Wikipedia articles in English(en) and Indic languages (Hindi(hi),

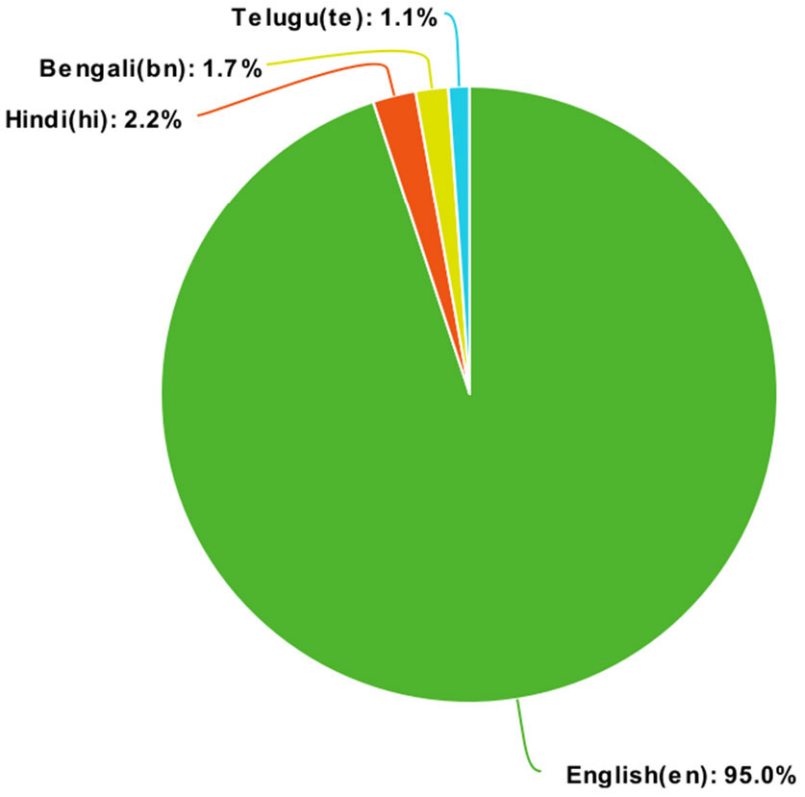

Figure 1. A pie chart comparing the counts of Wikipedia articles in English (en), Hindi (hi), Bengali (bn), and Telugu (te) languages.

Bengali(bn), and Telugu(te)).[a] Based on this analysis, it is apparent that the available monolingual corpora for LRLs (2.2 percent for Hindi, 1.7 percent for Bengali, and 1.1 percent for Telugu) are negligible compared to English (95.0 percent). The training of language models (LM) requires a huge corpus (Devlin *et al.* 2019; Conneau *et al.* 2020). This demands a need of developing LM learning techniques for LRLs that can be leveraged from RRLs.

The transfer-learning mechanism has improved the ability of LRLs when combined with multilingual models like mBERT (Devlin *et al.* 2019), XLM-R (Conneau *et al.* 2020), and IndicBERT (Kakwani *et al.* 2020). The authors (Artetxe *et al.* 2020) have shown that the monolingual models can provide multilingual-compatible performance for LRLs using unsupervised low-resource data and supervised task-specific data in RRLs. They have improved the performance of the monolingual models for LRLs to be comparable to the multilingual models. The other recent approaches to support LRLs in multilingual model environments include a transliteration of LRLs into the related prominent language (Khemchandani *et al.* 2021), utilizing overlapping tokens in the embedding learning of LRLs (Pfeiffer *et al.* 2021), model fine-tuning with expanded vocabulary (Wang, Mayhew, and Roth 2020), and light-weight adapter layer training with a fixed pre-trained model (Pfeiffer *et al.* 2020b). The impact of linguistic family and the use of RRL family members

[a]The article counts are taken from https://meta.wikimedia.org/wiki/List_of_Wikipedias on 12 November 2022. From the total articles of these 4 languages, 6,422,000+ are written in English(en), whereas the total Telugu(te) articles are 74,000+.

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

Table 1. Example question (ID: 57290ee2af94a219006aa002) along with the corresponding statement from the contextual paragraph containing the answer

Question: Hindi	यह कैसे निर्धारित किया गया कि प्रत्येक दलों से कितने नियुक्त होंगे?
Question: English	How was it determined how many from each camp would be appointed?
English statement with answer	Under the deal, the president would appoint cabinet ministers from both PNU and ODM camps depending on each party's strength in Parliament
Hindi statement	अनुबंध के अंतर्गत, राष्ट्रपति को प्रत्येक दल की संसद में शक्ति के आधार पर दोनों पीएनयू और ओडीएम दलों से कैबिनेट मंत्री नियुक्त करने थे। anubandh ke antargat, raashtrapati ko pratyek dal kee sansad mein shakti ke aadhaar par donon peeenayoo aur odeeem dalon se kaibinet mantree niyukt karane the
Bengali statement	চুক্তির অধীনে, রাষ্ট্রপতি সংসদে প্রতিটি দলের শক্তির উপর নির্ভর করে PNU এবং ODM উভয় শিবির থেকে মন্ত্রিপরিষদ মন্ত্রীদের নিয়োগ করবেন। Cuktira adhınē, rāṣṭrapati sansadē pratiṭi dalēra śaktira upara nirbhara karē PNU ēbaṁ ODM ubhaẏa śibira thēkē mantripariṣada mantrīdēra niẏōga karabēna
Telugu statement	ఈ ఒప్పందం ప్రకారం, పార్లమెంట్‌లో ప్రతి పక్షం బలాన్ని బట్టి PNU మరి-యు ODM క్యాంపుల నుండి క్యాబినెట్ మంత్రులను అధ్యక్షుడు నియమిస్తారు Ī oppandaṁ prakāraṁ, pārlamenṭlō prati pakṣaṁ balānni baṭṭi PNU mariyu ODM kyāmpula nuṇḍi kyābineṭ mantrulanu adhyakṣuḍu niyamistāru

Note: Words that appear in both the question and context are underlined, and similar introductory sentence structures are highlighted in red across all languages. The expected answer is highlighted with a yellow background.

for LRL acquisition demand further investigation. A distinctive departure in our methodology compared to the previously mentioned methodologies lies in our pursuit of a joint model training strategy across cognate languages, followed by task-specific refinement through fine-tuning using the same familial language. Additionally, our model not only leverages annotated data from the RRL but also incorporates data from the family-linked low-resource language (family-LRL) to enhance performance in downstream QA task.

The Indic languages, Hindi and Bengali, belong to the same language family and thus share the genealogical similarity of the Indo-Iranian group, whereas despite being part of the Dravidian language family, the embedding representation for Telugu (Tibeto-Burman group) is nearer to Hindi (Kudugunta *et al.* 2019) due to its typological similarity (Littell *et al.* 2017). Moreover, all three languages share the same subject-object-verb order in contrast to the subject-verb-object order of English. The evolutionary and geographical proximity generates high words sharing ratio (Khemchandani *et al.* 2021) among these languages vis-à-vis the word overlapping ratio with English. In this work, we have investigated the impact of Bengali and Telugu languages on Hindi zero-shot and few-shot question-answering.

To observe the language structure of the Indic languages, for the given question, we compared the sequence and the position of answer words in the context paragraph for all the languages. One such example from the XQuAD dataset is shown in Table 1. The answer words in Hindi are closer to the answer words of translated Bengali and Telugu statements. However, in the English language, the answer words are usually found at the end of the sentence. For all the languages, we have underlined the overlapping words between the question and context statement. We have also plotted parallel Hindi, Bengali, Telugu, and English words in the embedding space to see how far their embedding representations are. The observations and analysis section of our paper contains a detailed analysis of embedding distance charts.

In this article, we examine the suitability of using multilingual task-specific training to improve the performance of monolingual QA tasks. In the absence of sufficient supervised monolingual corpus, the performance of the system is not up to the mark. In order to solve the problem of small corpus size, we propose a way of utilizing the combination of an unsupervised corpus of more than

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

one language. To investigate the impact of combining the data of multiple languages and language relatedness, in this work, we show that, with small task-specific supervision of language from the same language family, the performance of multilingual models can be improved. Specifically, we have performed our experiments using QA data of three Indic languages (Hindi, Bengali, and Telugu) from TyDi (Clark *et al.* 2020), MLQA (Lewis *et al.* 2020), and XQuAD (Artetxe *et al.* 2020) datasets. Our experiments are conducted on pre-trained XLM-R, mBERT, and IndicBERT transformer models. For all the experiments we have used English as RRL.

Our major contributions are the following:

(1) To address the data scarcity problem in task-specific learning, we have proposed an idea of learning a task for LRL (Hindi) using supervised data of RRL (English) and supervised task-specific data of another LRL (Bengali/Telugu) which has structural and grammatical similarity with Hindi.
(2) In our experiments, we have analyzed the impact of the few-shot task learning sequence swapping between LRL (Hindi) and another LRL (Bengali/Telugu). This experiment enables us to observe the behavior and how the order of language impact overall learning while fine-tuning the model with multiple languages.
(3) The generalized nature of the proposed approach allows it to be extended for any transformer architecture. To verify this behavior, we conducted our experiments on XLM-R, mBERT, and IndicBERT, all of which follow different architectures.

2. Related work

The development of multilingual transformer models like mBERT (Devlin *et al.* 2019), XLM-R (Conneau *et al.* 2020), and IndicBERT (Kakwani *et al.* 2020) has shown significant performance gains on LRLs. The approach of decoupled encoding to identify related subwords is explored by Wang *et al.* (2019). The recent articles (Hu *et al.* 2020; Lauscher *et al.* 2020) revealed that cross-lingual transfer cannot be achieved by offering joint training for LRL and RRL due to the model's incapacity to accommodate several languages at the same time.

The research of linguistic relatedness begins with a translated parallel corpus and cross-lingual performance training based on concatenated parallel data (CONNEAU and Lample (2019)). Efforts have been made to overcome the ordinary performance due to poor translation (Goyal, Kumar, and Sharma 2020; Khemchandani *et al.* 2021; Song *et al.* 2020) by adopting the aspect of RRL transliteration to generate parallel data in LRL. Chung *et al.* (2020) proposed the direction of multilingual task learning using weighted clustered vocabulary. Cao, Kitaev, and Klein (2020) and Wu and Dredze (2020) explored altering the direction of contextual embedding by bringing the embedding of the aligned words closer together to achieve efficient cross-lingual transfer.

Wang *et al.* (2020) propose that monolingual LRL performance can be improved by increasing embedding layers with LRL-specific weights. Adding the language adapters and task adapters (Pfeiffer *et al.* 2020a, b; Houlsby *et al.* 2019) over transformer models to boost the performance of LRLs is the recent approach of performance improvement by small fine-tuning (Artetxe *et al.* 2020; Pandya, Ardeshna, and Bhatt 2021; Üstün *et al.* 2020). The approach of tokenizer learning for LRL and utilizing lexical overlap between LRL and RRL in embedding is adopted by Pfeiffer *et al.* (2021).

2.1 Embedding learning for the low-resource setup

For embedding learning, it is exceedingly difficult to obtain monolingual data for a majority of languages in the Indian language family. Hence, the monolingual embeddings for these languages are usually of poor quality (Michel, Hangya, and Fraser 2020). Eder, Hangya, and Fraser (2021)

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

suggested an embedding that starts with a small bilingual seed dictionary and pre-trained monolingual embeddings of the RRLs. Adams *et al.* (2017) demonstrated that training monolingual embedding for LRLs and RRLs together improves the monolingual embedding quality of LRLs. Lample *et al.* (2018b) trained the fastText skipgram embeddings (Bojanowski *et al.* 2017) to learn the joint embedding of the source and the target languages using joint corpora. Vulić *et al.* (2019) showed that the unsupervised approach (Artetxe, Labaka, and Agirre 2018) cannot efficiently handle LRLs and multiple distant.

Using adversarial training, Zhang *et al.* (2017) demonstrated that monolingual alignment is possible without bilingual data. Lample *et al.* (2018a) combined the Procrustes analysis refinement and adversarial training to obtain an unsupervised mapping. The bottleneck with the mapping approaches lies in its dependence on high-quality monolingual embedding spaces. The approach reported by Wang *et al.* (2020) benefits from the conjunction of joint and mapping methods, where initially they trained combined monolingual datasets. Subsequently, they map the source and target embedding after reallocation of the oversharing vocabularies.

2.2 Language relatedness

The idea of using RRLs to improve LRLs is to reduce the need for supervised data in the LRL. The authors (Nakov and Ng 2009) have proposed the statistical machine translation model, which requires a few parallel samples of source LRL and target RRL in addition to a large parallel corpus of target RRL and another RRL, which is related to the target RRL. During the transfer learning from RRL to LRL, Nguyen and Chiang (2017) exploited the shared word embeddings.

Until now, only a few works have considered using information from related RRLs for low-resource embeddings (Woller, Hangya, and Fraser 2021). Many researchers have looked at the idea of joint training, which either necessitates a huge training corpus or is reliant on pre-trained monolingual embedding (Ammar *et al.* 2016; Alaux *et al.* 2019; Chen and Cardie 2018; Devlin *et al*. 2019; Heyman *et al.* 2019).

A major distinction in our approach from the above approaches is that we have looked into the direction of joint model training on related languages followed by task-specific fine-tuning with the family language. Instead of independently addressing individual languages, our approach capitalizes on the inherent linguistic relationships between related languages. By collectively training a model across related languages, we harness the potential for cross-lingual knowledge transfer, allowing the model to develop a more comprehensive understanding of underlying linguistic structures and nuances shared within the language family. This approach aligns with the linguistic theory that languages with common ancestry exhibit similarities in syntax, semantics, and other linguistic attributes. Furthermore, the subsequent task-specific fine-tuning using the same family language aligns with the idea that linguistic features and patterns learned during the joint training phase can be further refined to suit the specific tasks at hand. This two-step process not only capitalizes on the advantages of cross-lingual training but also tailors the model to effectively address domain-specific challenges within the context of the selected family language.

In addition, our approach depends on MLM training using family languages to establish the word correlation between family languages. Here, both languages are of the LRL category; hence the joint MLM training before fine-tuning on downstream task and customized learning framework obviates the requirement for corpus translation or transliteration, thereby reducing the laborious processes of data labeling, and the adjustment of start and end tokens inherent in standard QA-supervised datasets.

3. Proposed approach of cross-lingual language learning

This section describes our approach to transfer knowledge from one language (L2, Bengali/Telugu in this case) to another (L1, Hindi in this case). Here, L1 and L2 both fall under the category of

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

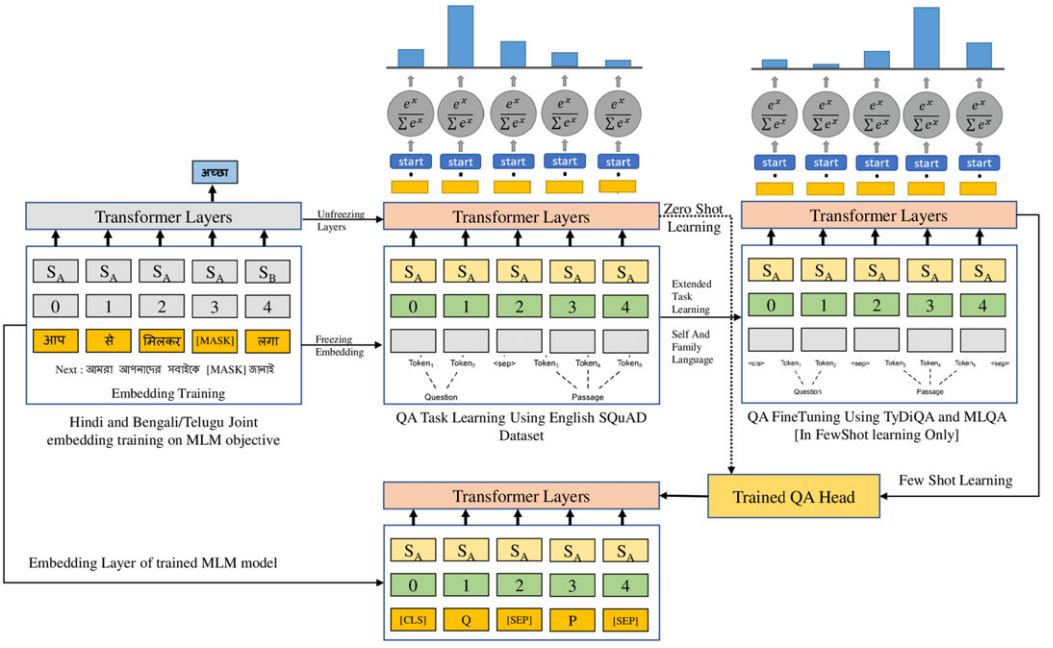

Figure 2. Proposed approach for low-resource question-answering.

LRLs with monolingual unsupervised datasets available. Additionally, a training corpus for the downstream task is also available for L2. The proposed method is summarized in the following steps and explained in detail in the subsequent paragraphs:

(1) Joint training of an embedding layer of pre-trained transformer models on L1 and L2 with masked language modeling (MLM) objective and unlabeled text corpora.[b] Here, pre-training is heavily biased on RRLs (L3). During the MLM training, except embedding, all other layers are kept frozen.

(2) Fine-tune the model on downstream tasks using labeled data in L3 and keeping embedding frozen. The result of this phase is that the model retains the acquired lexical representations without alterations, while also improving its capacity to obtain precise answers.

(3) Further fine-tune the model on downstream tasks using labeled data of L2 and L1. During this step, the embedding layer is again kept frozen. Here, we have analyzed the impact of changing the learning sequence from L2-L1 to L1-L2 in a few-shot setup.

(4) In a transfer-learning configuration, replace the embedding layer of the above setup with the embedding layer learned in step (1).

(5) Evaluate the model performance of L1 on downstream task using a test dataset of L1.

As shown in Fig. 2, we trained an embedding layer of a pre-trained transformer model with an MLM objective. During this training phase, the model is provided the input token vectors with some of the tokens randomly replaced with a specific (MASK) token. The objective here is for the model to predict the most suitable token from the vocabulary for the (MASK) token. To comprehend the language statistically, it is helpful to employ models that are trained utilizing MLM objective. We performed MLM training as the first step in our architecture, to get benefited from the nearness of similar token representation in a language family. Specifically, for

[b]Here, we have used English MLM pre-trained models from the HuggingFace: https://huggingface.co/.

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

our customized training architecture, during step (1), unsupervised data of Hindi (L1) and one of the family languages (Bengali/Telugu) (L2) are supplied with a 15 percent masking probability. Random statements from L1 and L2 are given to the model during MLM training to help it understand the commonality between these language families. Except for embedding, weights of other layers are kept unchanged as the objective here is to learn the embedding representation of the language family.

To learn the question-answering task, in step (2), we have fine-tuned our model using the SQuAD English dataset (Rajpurkar *et al.* 2016). The objective of this stage is to preserve the acquired lexical representations in an unaltered state while concurrently augmenting the model's capacity for obtaining the accurate answers.

For the zero-shot learning setup in step (3), the fine-tuning is performed on L2 TyDi Bengali/Telugu dataset (Clark *et al.* 2020). In a few-shot setup, we have further fine-tuned the QA learning using the MLQA Hindi dataset (Lewis *et al.* 2020). The rationale behind the implementation of this step is to strike a balance between capitalizing on the model's existing downstream task knowledge gained through the English QA learning (step (2)) and refining its linguistic proficiency in answering the question using family language L2.

In all QA learning steps, the embedding layer was kept frozen, that is, only all transformer layer parameters got updated except for the embedding layer during this task learning phase. Hence, the word-relatedness learned during step (1) would not be affected by the task learning phase.

At the end of step (3), in zero-shot setup, we have (i) an embedding layer inclined toward languages relatedness of L1 and L2 and (ii) a transformer model fine-tuned for a downstream task using L3 and L2. Both these layers are combined to measure the performance of the target LRL language. So, during the evaluation phase, in step (4), the embedding layer of the model trained on the QA task is replaced with the embedding layer trained in the first step. The new transferred architecture is evaluated on the Hindi test dataset.[c]

In all our experiments, the special transformer symbols ((CLS), (SEP), (UNK), and (MASK)) are shared among all languages.[d]

3.1 Models

We conducted our experiments on XLM-R, mBERT, and IndicBERT to determine the impact of Bengali and Telugu on Hindi. (All three languages are part of the Indian subcontinent and categorized as LRLs.) The mBERT model is pre-trained on 104 languages, whereas XLM-R is pre-trained on 100 languages. The training set of both includes Hindi, Bengali, and Telugu languages. IndicBERT is a multilingual ALBERT model that has been trained in a total of twelve Indian languages, including Hindi, Bengali, and Telugu. The reason for including the IndicBERT model in our study is that it has already been trained in a variety of Indian languages. Due to this, the influence of Indian-continent languages seems to be higher compared to the influence of RRLs on IndicBERT. As a result, the model's lexicon is dominated by Indian languages. We have trained the following models for zero-shot setup using XLM-R_{base},XLM-R_{Large} mBERT, and IndicBERT:

- MODEL-xxMLM-SQuAD[e]: The model is trained on a Wikipedia dump[f] of Hindi and Bengali/Telugu languages on MLM objective followed by task learning using the SQuAD dataset.

[c]Our code and the link to trained models can be accessed at https://github.com/pandyahariom/Low-Resource-QA.

[d]The special symbols are task-dependent, and downstream fine-tuning is performed on a different language than the one used during the transfer learning.

[e]Here, MODEL is one of the XLM-R_{base}, XLM-R_{Large}, mBERT, or IndicBERT, and xx is either bn(Bengali) or te(Telugu).

[f]The wikiextractor Attardi (2015) tool is used to extract raw text from Wikipedia dump.

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

- MODEL-xxMLM-SQuAD-TyDi : The MODEL-xxMLM-SQuAD model is further trained using one of the LRLs (L2) as mentioned in the proposed approach.
- MODEL-xxMLM-SQuAD-TyDi-NotFreeze : The model training and parameters are the same as in MODEL-xxMLM-SQuAD-TyDi, with the exception that the embedding parameters are not frozen during TyDiQA training. The reason for using this paradigm in our research is that the LRL training focuses on a member of the target language's language family; combining it with embedding training with task learning can enhance performance. However, in the performance, we have not observed a significant impact of this change. So, we have excluded this step in the training of few-shot learning models.

For the few-shot learning, we have used Hindi dataset from MLQA (Lewis *et al.* 2020) in addition to the datasets used in the zero-shot setup. We have trained the following models for a few-shot setup using XLM-R_{Large}, mBERT, and IndicBERT:

- MODEL-xxMLM-SQuAD-MLQA[g]: The model is trained on a Wikipedia dump of Hindi and Bengali/Telugu languages on MLM objective followed by task learning using the SQuAD dataset and few-shot task learning on the MLQA dataset.
- MODEL-xxMLM-SQuAD-MLQA-TyDi: MODEL-xxMLM-SQuAD model further trained using one of the LRLs (L2) as mentioned in the proposed approach.
- MODEL-xxMLM-SQuAD-TyDi-MLQA: In this training setup, the arrangement of few-shot learning and language family learning is reverse of the above model.

4. Experimental setup

4.1 Model setup

For training our model, we use Adam optimizer with learning rate as 2e−5 and adam_epsilon = 1e−8. All the fine-tunings are performed with a single epoch and batch size 4. Common values for all the transformer architecture include warmup_proportion = 0.1, weight_decay = 0.01, intialize_range = 0.02, max_position_embeddings = 512, hidden_act = glue, position_embedding_type = absolute, max_seq_length for MLM = 128, max_seq_length for QA = 384, doc_stride = 128, n_best_size = 20, and max_answer_length = 30. In addition to these hyperparameters, Table 2 indicates the architecture-specific values of other hyperparameters.

Additionally, by following the standard QA training setup, we are truncating context only when the combined length of question and context is going beyond the model size. The question with the remaining tokens of context will be given to the model in the next training sample. Along with all predictions, we are returning offset mapping to map token indices. It enables us to find the end token position depending on the predicted start token and the length of the answer.

4.2 Datasets

The datasets used in our experiments are categorized into two categories: (1) unlabelled text data for MLM objective and (2) question-answering data in the SQuAD format. In this section, the details of our dataset are mentioned for both categories:

4.2.1 Unlabelled data for MLM objective

To perform the embedding learning for low-resource languages (Hindi, Bengali, and Telugu in our case), we have combined Wikipedia dump, IndicCorp (Kakwani *et al.* 2020), and LRL part from *Samanantar* parallel Indic corpora collection (Ramesh *et al.* 2021). Here, in *Samanantar*,

[g]Here, MODEL is one of the XLM-R_{Large}, mBERT, or IndicBERT, and xx is either bn(Bengali) or te(Telugu).

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

Table 2. Hyperparameters for fine-tuning our models (XLM-R, mBERT, and IndicBERT) on MLM and QA tasks

Hyperparameter	IndicBERT	mBERT	XLM-R
layer_norm_eps	1e−12	1e−12	1e−05
attention_probs_dropout_prob	0	0.1	0.1
num_attention_heads	12	12	16
hidden_size	768	768	1,024
num_hidden_layers	12	12	24

Table 3. Total number of sentences used per LRL for the MLM fine-tuning

#Sentences:	IndicCorp	Samanantar	Wikipedia dump	Total
Hindi(hi)	63.1M	8.56M	2.3M	73.96M
Bengali(bn)	39.9M	8.52M	1.45M	49.87M
Telugu(te)	47.9M	4.82M	1.39M	54.11M

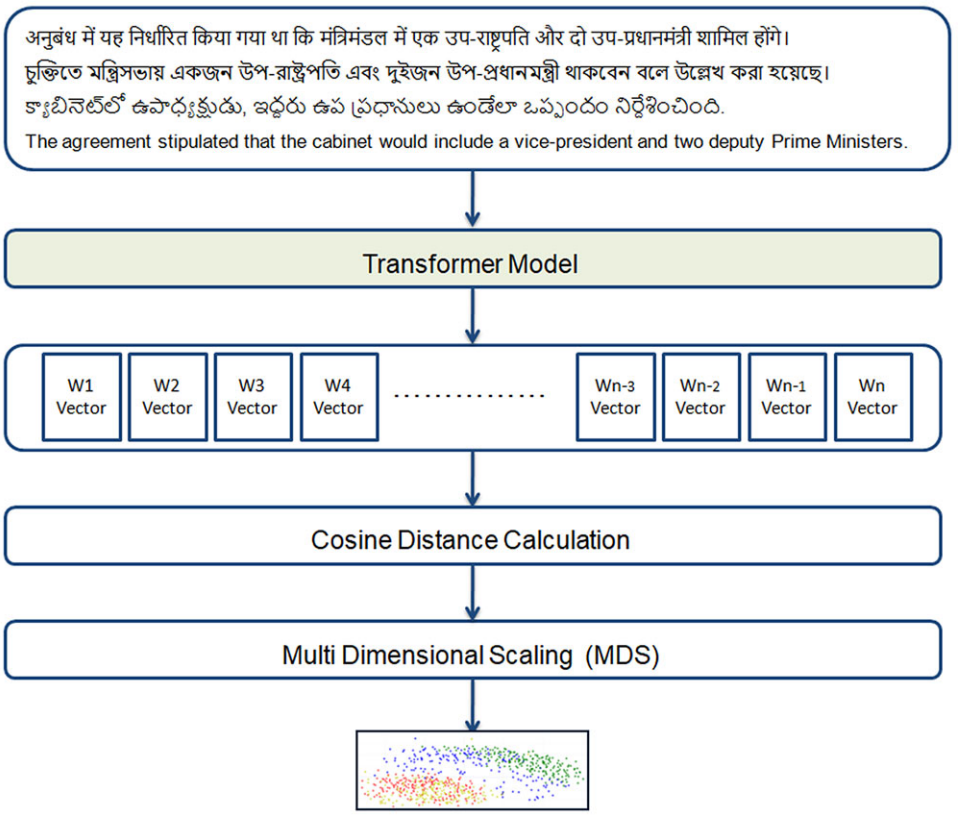

Figure 3. Visual representation of embedding of a context paragraph from all LRLs and RRL.

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

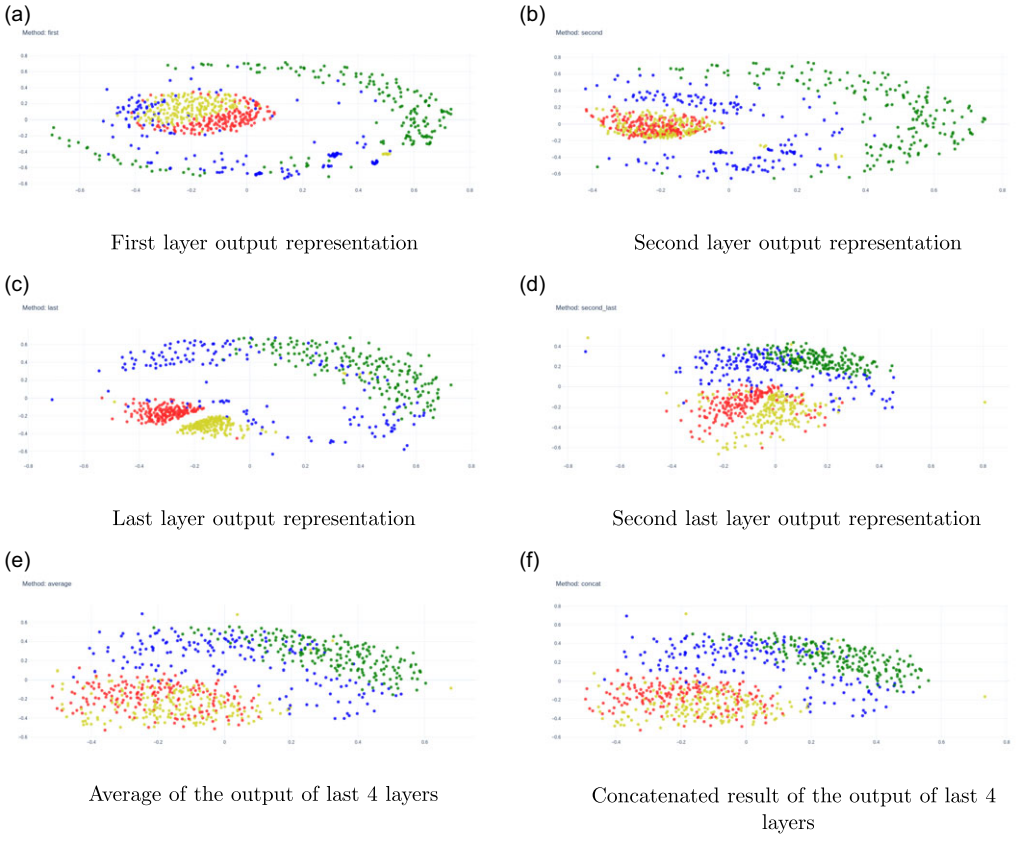

Figure 4. The embedding representation of context data of Hindi, Bengali, Telugu and English parallel data. The interpretation of colors—yellow:Bengali text, red:Hindi text, blue:Telugu text, and green:English text.

parallel data for Indic languages and English is given. *Samanantar* also contains parallel data between Indic languages. From that dataset, we have used individual (without a parallel corpus) Hindi, Bengali, and Telugu data and combined it with Wikipedia dump and IndicCorp for each language. The total number of sentences per LRL language used for our experiments is shown in Table 3.

4.2.2 Task-specific data for question-answering objective

Initial task-specific training is performed on RRL (English) SQuAD 1.1 (Rajpurkar *et al.* 2016). To train the model further on LRL downstream task, we have used *TyDiQA secondary task* (Clark *et al.* 2020) dataset for Bengali (bn) or Telugu (te) languages depending on the chosen language of family-LRL. In the few-shot setup, the MLQA dataset (Lewis *et al.* 2020) is used to train the model on Hindi (hi) QA task. All of our models are evaluated on the XQuAD (Artetxe, Ruder, and Yogatama 2020) Hindi dataset which contains 240 paragraphs and 1,190 question-answer pairs.

5. Observations and analysis

This section encompasses our empirical investigation pertaining to lexical representations of parallel data across chosen LRL languages (Hindi, Bengali and Telugu) and RRL (English),

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

Table 4. F1 score and EM of models on the XQuAD-Hindi dataset after few-shot learning on the Hindi MLQA dataset

Previous models	F1	EM		
mBERT†	59.2	46.0	–	–
XLM-$R^{\dagger}_{Large}$	76.7	59.7	–	–
Our models	Training with (bn)		Training with (te)	
	F1	EM	F1	EM
Indic-xxMLM-SQuAD-MLQA	39.53	21.77	39.06	22.18
Indic-xxMLM-SQuAD-MLQA-TyDi	35.36	17.48	33.17	12.44
Indic-xxMLM-SQuAD-TyDi-MLQA	39.84	22.19	39.33	22.35
mBERT-xxMLM-SQuAD-MLQA	69.87	54.21	69.62	52.45
mBERT-xxMLM-SQuAD-MLQA-TyDi	65.62	49.50	65.79	51.60
mBERT-xxMLM-SQuAD-TyDi-MLQA	70.06	53.87	69.62	53.36
XLM-R_{Large}-xxMLM-SQuAD-MLQA	78.64	62.53	78.77	61.79
XLM-R_{Large}-xxMLM-SQuAD-MLQA-TyDi	75.50	57.82	74.47	57.06
XLM-R_{Large}-xxMLM-SQuAD-TyDi-MLQA	80.54	64.12	79.74	62.44
	MLM training on unlabeled Hindi data			
Indic-hiMLM-SQuAD-MLQA	38.14	21.03		
mBERT-hiMLM-SQuAD-MLQA	67.38	53.21		
XLM-R_{Large}-hiMLM-SQuAD-MLQA	77.98	61.87		

Note: "xx" indicates the fine-tuning language (be/te), as specified in the column header. Models are trained jointly on Hindi+Bengali or Hindi+Telugu, followed by QA task learning. Results with † are taken from the XQuAD paper (Artetxe *et al*. 2020) for comparison purpose.

accompanied by a discourse on the outcomes acquired through both the few-shot and zero-shot setups with family-LRL (Bengali/Telugu).

5.1 Analysis of the embedding representation of corresponding terms

To analyze the embedding representation of parallel words of Hindi, Bengali, Telugu, and English, we have randomly chosen a parallel Hindi and English context paragraph from the XQuAD dataset. For generating Telugu and Bengali parallel statements, we have translated the Hindi context paragraph. The authors (Abnar and Zuidema 2020) have analyzed that information originating from different tokens gets increasingly mixed across the layers of the transformer. Hence, instead of looking at only the raw attention in a particular layer, we have considered a weighted flow of information from the input embedding to the particular hidden output.

To interpret the effect of specific word representation, we have extracted output vectors of the first, second, penultimate, and last layers. Additionally, we have calculated the average and concatenated vectors from the last four layers to retrieve combined vectors. Next, we calculated the cosine distance between them in the higher dimension to observe the proximity of these vectors. Finally, using the multidimensional scaling technique, we have plotted the distance matrix. Figure 3 represents our approach to data representation.

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

Table 5. Zero-shot results of F1 score and EM on XQuAD-Hindi dataset

Previous models	F1	EM		
mBERT†	59.2	46.0	–	–
XLM-$R^{\dagger}_{Large}$	76.7	59.7	–	–
Our models	Training with (bn)		Training with (te)	
	F1	EM	F1	EM
Indic-xxMLM-SQuAD	26.83	12.44	19.25	7.48
Indic-xxMLM-SQuAD-TyDi	24.48	10.17	23.27	7.56
Indic-xxMLM-SQuAD-TyDi-NotFreeze	24.28	10.07	22.17	6.89
mBERT-xxMLM-SQuAD	57.87	43.53	56.10	43.61
mBERT-xxMLM-SQuAD-TyDi	54.23	39.66	53.97	39.92
mBERT-xxMLM-SQuAD-TyDi-NotFreeze	53.06	39.16	53.87	39.84
XLM-R_{base}-xxMLM-SQuAD	68.22	52.27	69.03	52.77
XLM-R_{base}-xxMLM-SQuAD-TyDi	63.44	45.88	63.90	45.80
XLM-R_{base}-xxMLM-SQuAD-TyDi-NotFreeze	63.28	45.88	64.30	46.72
XLM-R_{Large}-xxMLM-SQuAD	76.08	60.18	75.81	60.17
XLM-R_{Large}-xxMLM-SQuAD-TyDi	71.63	52.52	73.40	56.72
XLM-R_{Large}-xxMLM-SQuAD-TyDi-NotFreeze	71.47	52.27	73.18	56.48
	MLM training on unlabeled Hindi data			
Indic-hiMLM-SQuAD	25.54	11.87		
mBERT-hiMLM-SQuAD	55.28	42.18		
XLM-R_{base}-hiMLM-SQuAD	67.29	50.62		
XLM-R_{Large}-hiMLM-SQuAD	74.89	58.99		

Note: "xx" indicates the MLM fine-tuning language (be/te), as specified in the column header. Models undergo the MLM fine-tuning jointly on Hindi+Bengali or Hindi+Telugu followed by QA task learning. Results with † are taken from original XQuAD paper (Artetxe *et al.* 2020) for comparison purpose.

Figure 4 indicates the representation results obtained using the mBERT model, which was trained on an unsupervised Hindi+Bengali dataset with an MLM objective. Here, the average and concatenated result shows that, due to the joint training and language proximity, the distance between the Bengali and Hindi words is less compared to Telugu and English. Moreover, despite Telugu language was not given in MLM training, the representation of Telugu words is nearer compared to English. This result also adds the attestation to the Indic family languages and relatedness concept.

5.2 Performance observation in the context of utilizing Bengali as the family language

Our observations for the few-shot setup are shown in Table 4. While considering Bengali as the family language, we have noted that few-shot learning followed by language family learning improves the benchmark F1 score of mBERT by 6.42 percent. However, by changing the sequence of language family learning and few-shot learning, the improvement in the F1 score is 10.86

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

Table 6. F1 score and EM of models on MLQA Hindi test dataset in the few-shot configuration

Our models	Training with (bn)		Training with (te)	
	F1	EM	F1	EM
Indic-xxMLM-SQuAD-MLQA	45.02	26.46	44.00	25.73
Indic-xxMLM-SQuAD-TyDi-MLQA	45.02	26.33	44.92	26.29
mBERT-xxMLM-SQuAD-MLQA	74.89	57.79	74.94	57.60
mBERT-xxMLM-SQuAD-TyDi-MLQA	75.41	58.19	75.17	58.11
XLM-R_{Large}-xxMLM-SQuAD-MLQA	77.88	60.78	81.14	64.31
XLM-R_{Large}-xxMLM-SQuAD-TyDi-MLQA	80.95	63.74	80.81	63.63

Note: "xx" indicates the fine-tuning language (be/te), as specified in the column header. Models are trained jointly on Hindi+Bengali or Hindi+Telugu followed by QA task learning.

Table 7. F1 score and EM of models on TyDiQA test dataset in few-shot configuration

Our models	Training with (bn)		Training with (te)	
	F1	EM	F1	EM
Indic-xxMLM-SQuAD-TyDi	39.51	23.01	69.22	54.41
Indic-xxMLM-SQuAD-MLQA-TyDi	41.75	20.35	69.13	53.51
mBERT-xxMLM-SQuAD-TyDi	74.59	66.38	83.84	70.25
mBERT-xxMLM-SQuAD-MLQA-TyDi	74.83	61.95	83.95	70.41
XLM-R_{Large}-xxMLM-SQuAD-TyDi	81.68	69.91	85.16	71.30
XLM-R_{Large}-xxMLM-SQuAD-MLQA-TyDi	83.95	72.57	85.31	71.15

Note: "xx" indicates fine-tuning language, as specified in the column header. Models are trained jointly on Hindi+Bengali or Hindi+Telugu followed by QA task learning.

percent, which is 4.44 percent better than the preceding setup. Similar results are obtained for XLM-R_{Large} where the language family learning followed by a few-shot improves the performance by 5.04 percent compared to the reverse learning setup.

5.3 Performance observation in the context of utilizing Telugu as the family language

While considering Telugu as the family language, the sequence family learning followed by a few-shot improves the mBERT result by 1.76 percent and XLM-R_{Large} result by 5.38 percent. All these analyses show the importance of keeping a few-shot setup at the end of fine-tuning. It is better in tuning the embedding parameter toward the target language. We have also observed that in the zero-shot setup, the joint MLM training helps in achieving performance comparable to the benchmark, even if the task training is achieved using a single epoch of the SQuAD dataset. However, the additional task learning using the family language dataset from TyDiQA degrades the performance of the architecture. Table 5 shows our results on IndicBERT, mBERT, and XLM-R models using Bengali and Telugu languages.

Table 6 indicates the result obtained on the MLQA dataset in the few-shot learning setup. Much similar to the previous case, our combined model shows promising results in the target language. Table 7 shows the performance of our models on the TyDiQA Bengali and Telugu testsets. The results show the positive influence of Hindi + Bengali and Hindi + Telugu MLM on the performance of Bengali/Telugu languages.

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

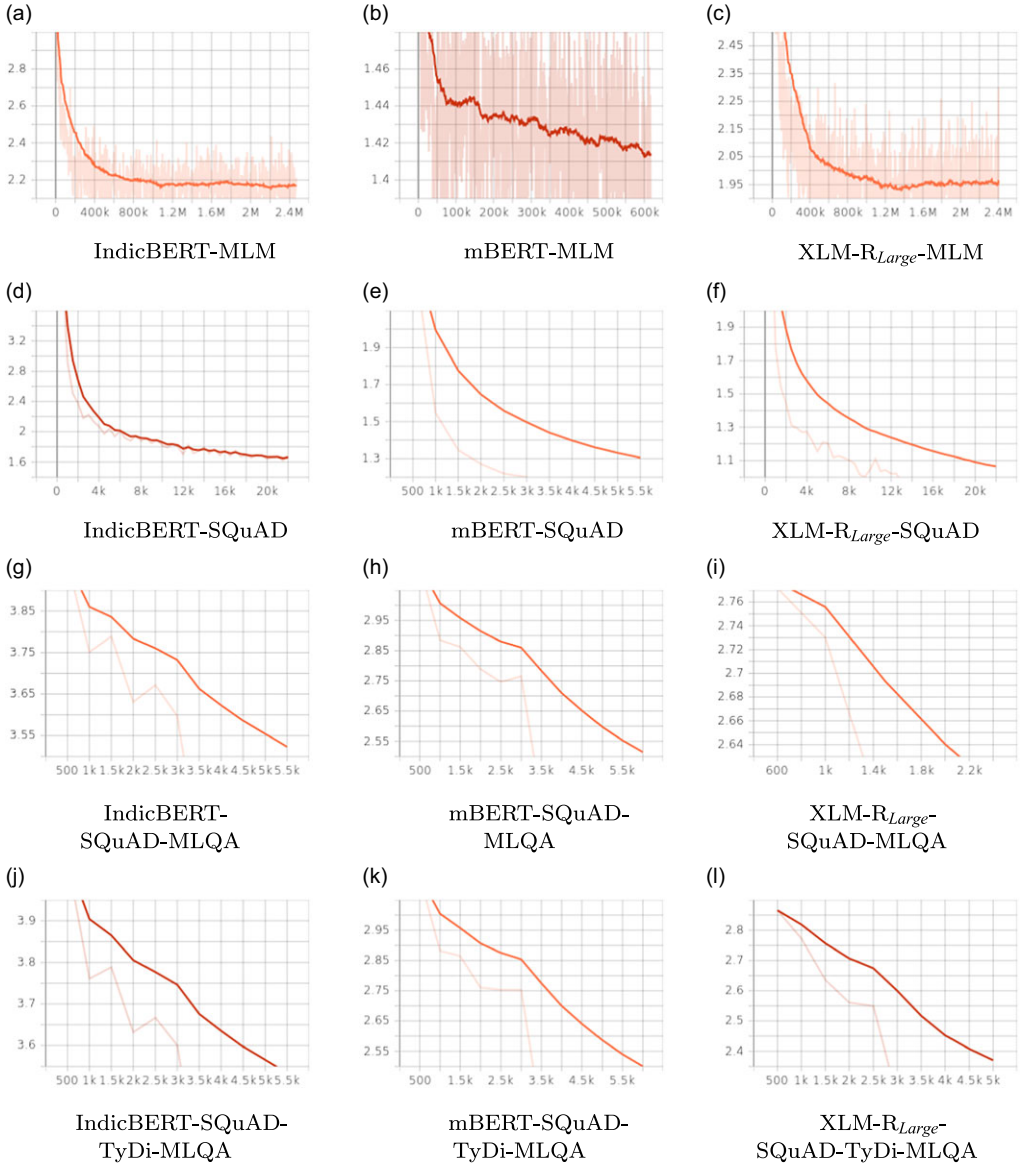

Figure 5. Training loss at each stage of the few-shot learning steps on IndicBERT, mBERT, and XLM-R$_{Large}$ models on Hindi (hi) + Bengali(bn) joint datasets with smoothing = 0.985.

Figure 5 indicates the training loss of our IndicBERT, mBERT, and XLM-R$_{Large}$ models. For all three models, training loss indicates the few-shot learning impact of all the steps mentioned in our proposed approach in Section 3.

6. Conclusion

In this article, we proposed an approach to train the model in LRL using supervised data from English and another language belonging to the same family, particularly, for QA task. Our custom learning approach has shown enhanced performance for Hindi LRL when QA fine-tuning

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

is carried out in Bengali and Telugu languages in addition to the task learning using English RRL supervised data. Moreover, the results of our experiments on XLM-R_{Large}, mBERT, and IndicBERT transformer models show that in case of unavailability of parallel data for LRL, joint MLM training with other LRL family language followed by task learning with RRL and family-LRL improves few-shot performance for LRL. However, task learning only using family-LRL seems inefficient due to the limited amount of task-specific data availability in family-LRL.

In a few-shot setup, our best performing XLM-R_{Large} model has achieved 80.54/64.12 (F1 score/EM), while Bengali is used as a family language, whereas 79.74/62.44 (F1 score/EM) is observed with Telugu training. For zero-shot setup, 76.08/60.18 (F1 score/EM) and 75.81/60.17 (F1 score/EM) are the results obtained with Bengali and Telugu, respectively. Using mBERT model in a few-shot setup, 70.06/53.87 (F1 score/EM) is the best score when Bengali language is used as a family language, whereas 69.62/53.36 (F1 score/ EM) is observed with Telugu training. For mBERT in zero-shot setup, 57.87/43.53 (F1 score/EM) and 56.10/43.61 (F1 score/EM) are the results of Bengali and Telugu, respectively. The improvement in the results justifies that with the proposed custom learning approach, learning from the language family is indeed helpful.

Although it is not explored in this paper, we believe that the concept of learning from a language family can be applied to other LRLs as well. The direction of adopting the family-learning technique to other downstream tasks is an avenue for future research.

References

Abnar S. and Zuidema W. (2020). Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 4190–4197, Online.

Adams O., **Makarucha A.**, **Neubig G.**, **Bird S. and Cohn T.** (2017). Cross-lingual word embeddings for low-resource language modeling. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, Valencia, Spain. Association for Computational Linguistics, pp. 937–947.

Alaux J., **Grave E.**, **Cuturi M. and Joulin A.** (2019). Unsupervised hyper-alignment for multilingual word embeddings. In *International Conference on Learning Representations*.

Ammar W., **Mulcaire G.**, **Tsvetkov Y.**, **Lample G.**, **Dyer C. and Smith N.A.** (2016). Massively multilingual word embeddings.

Artetxe M., **Labaka G. and Agirre E.** (2018). A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia. Association for Computational Linguistics, pp. 789–798.

Artetxe M., **Ruder S. and Yogatama D.** (2020). On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, pp. 4623–4637, Online.

Attardi G. (2015). Wikiextractor. Available at https://github.com/attardi/wikiextractor.

Bojanowski P., **Grave E.**, **Joulin A. and Mikolov T.** (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* **5**, 135–146.

Brown T., **Mann B.**, **Ryder N.**, **Subbiah M.**, **Kaplan J.D.**, **Dhariwal P.**, **Neelakantan A.**, **Shyam P.**, **Sastry G.**, **Askell A.**, **Agarwal S.**, **Herbert-Voss A.**, **Krueger G.**, **Henighan T.**, **Child R.**, **Ramesh A.**, **Ziegler D.**, **Wu J.**, **Winter C.**, **Hesse C.**, **Chen M.**, **Sigler E.**, **Litwin M.**, **Gray S.**, **Chess B.**, **Clark J.**, **Berner C.**, **McCandlish S.**, **Radford A.**, **Sutskever I. and Amodei D.** (2020). Language models are few-shot learners. In Larochelle H., Ranzato M., Hadsell R., Balcan M. and Lin H. (eds), *Advances in Neural Information Processing Systems*, vol. **33**. Curran Associates, Inc., pp. 1877–1901.

Cao S., **Kitaev N. and Klein D.** (2020). Multilingual alignment of contextual word representations. In *International Conference on Learning Representations*.

Chen X. and Cardie C. (2018). Unsupervised multilingual word embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium. Association for Computational Linguistics, pp. 261–270.

Chung H.W., **Garrette D.**, **Tan K.C. and Riesa J.** (2020). Improving multilingual models with language-clustered vocabularies. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, pp. 4536–4546, Online.

Clark J.H., **Choi E.**, **Collins M.**, **Garrette D.**, **Kwiatkowski T.**, **Nikolaev V. and Palomaki J.** (2020). TyDi QA: a benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics* **8**, 454–470.

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L. and Stoyanov V. (2020). Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 8440–8451, Online.

Conneau A. and Lample G. (2019). Cross-lingual language model pretraining. In Wallach H., Larochelle H., Beygelzimer A., d' Alché-Buc F., Fox E. and Garnett R. (eds), *Advances in Neural Information Processing Systems*, vol. **32**. Curran Associates, Inc.

Devlin J., Chang M.-W., Lee K. and Toutanova K. (2019). Bert: pre-training of deep bidirectional transformers for language understanding.

Eder T., Hangya V. and Fraser A. (2021). Anchor-based bilingual word embeddings for low-resource languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Association for Computational Linguistics, pp. 227–232, Online.

Goyal V., Kumar S. and Sharma D.M. (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*. Association for Computational Linguistics, pp. 162–168, Online.

Heyman G., Verreet B., Vulić I. and Moens M.-F. (2019). Learning unsupervised multilingual word embeddings with incremental multilingual hubs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota. Association for Computational Linguistics, vol. **1**, pp. 1890–1902.

Houlsby N., Giurgiu A., Jastrzebski S., Morrone B., De Laroussilhe Q., Gesmundo A., Attariyan M. and Gelly S. (2019). Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*. PMLR, vol. **97**, pp. 2790–2799.

Hu J., Ruder S., Siddhant A., Neubig G., Firat O. and Johnson M. (2020). Xtreme: a massively multilingual multi-task benchmark for evaluating cross-lingual generalization. CoRR, abs/2003.11080.

Kakwani D., Kunchukuttan A., Golla S., G. N.C., Bhattacharyya A., Khapra M.M. and Kumar P. (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*. Association for Computational Linguistics, pp. 4948–4961, Online.

Kaur P., Pannu H.S. and Malhi A.K. (2021). Comparative analysis on cross-modal information retrieval: a review. *Computer Science Review* **39**, 100336. https://www.sciencedirect.com/science/article/pii/S1574013720304366

Khemchandani Y., Mehtani S., Patil V., Awasthi A., Talukdar P. and Sarawagi S. (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)*. Association for Computational Linguistics, pp. 1312–1323, Online.

Kudugunta S., Bapna A., Caswell I. and Firat O. (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)*, Hong Kong, China. Association for Computational Linguistics, pp. 1565–1575.

Lample G., Conneau A., Ranzato M., Denoyer L. and Jégou H. (2018a). Word translation without parallel data. In *International Conference on Learning Representations*.

Lample G., Ott M., Conneau A., Denoyer L. and Ranzato M. (2018b). Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium. Association for Computational Linguistics, pp. 5039–5049.

Lauscher A., Ravishankar V., Vulić I. and Glavaš G. (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)*. Association for Computational Linguistics, pp. 4483–4499, Online.

Lewis P., Oguz B., Rinott R., Riedel S. and Schwenk H. (2020). MLQA: evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 7315–7330, Online.

Littell P., Mortensen D. R., Lin K., Kairis K., Turner C. and Levin L. (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*, Valencia, Spain. Association for Computational Linguistics, vol. **2**, pp. 8–14.

Liu J., Chen Y. and Xu J. (2022). Document-level event argument linking as machine reading comprehension. *Neurocomputing* **488**, 414–423.

Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L. and Stoyanov V. (2019). Roberta: A robustly optimized BERT pretraining approach. CoRR, abs/1907.11692.

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

Michel L., **Hangya V. and Fraser A.** (2020). Exploring bilingual word embeddings for Hiligaynon, a low-resource language. In *Proceedings of the 12th Language Resources and Evaluation Conference*, Marseille, France. European Language Resources Association, pp. 2573–2580.

Nakov P. and Ng H.T. (2009). Improved statistical machine translation for resource-poor languages using related resource-rich languages. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, Singapore. Association for Computational Linguistics, pp. 1358–1367.

Nguyen T.Q. and Chiang D. (2017). Transfer learning across low-resource, related languages for neural machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Taipei, Taiwan. Asian Federation of Natural Language Processing, pp. 296–301.

Ofoghi B., **Mahdiloo M. and Yearwood J.** (2022). Data envelopment analysis of linguistic features and passage relevance for open-domain question answering. *Knowledge-Based Systems* **244**, 108574.

Pandya H., **Ardeshna B. and Bhatt B.** (2021). Cascading adaptors to leverage English data to improve performance of question answering for low-resource languages. In *Proceedings of the 18th International Conference on Natural Language Processing (ICON)*, National Institute of Technology Silchar, Silchar, India. NLP Association of India (NLPAI), pp. 544–549.

Pandya H.A. and Bhatt B.S. (2021). Question answering survey: directions, challenges, datasets, evaluation matrices. CoRR, abs/2112.03572.

Pfeiffer J., **Rücklé A.**, **Poth C.**, **Kamath A.**, **Vulić I.**, **Ruder S.**, **Cho K. and Gurevych I.** (2020a). AdapterHub: a framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, pp. 46–54, Online.

Pfeiffer J., **Vulić I.**, **Gurevych I. and Ruder S.** (2020b). MAD-X: an adapter-based framework for multi-task cross-lingual transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, pp. 7654–7673, Online.

Pfeiffer J., **Vulić I.**, **Gurevych I. and Ruder S.** (2021). UNKs everywhere: adapting multilingual language models to new scripts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Online and Punta Cana, Dominican Republic*. Association for Computational Linguistics, pp. 10186–10203.

Raffel C., **Shazeer N.**, **Roberts A.**, **Lee K.**, **Narang S.**, **Matena M.**, **Zhou Y.**, **Li W. and Liu P.J.** (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* **21**(140), 1–67.

Rajpurkar P., **Zhang J.**, **Lopyrev K. and Liang P.** (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas. Association for Computational Linguistics, pp. 2383–2392.

Ramesh G., **Doddapaneni S.**, **Bheemaraj A.**, **Jobanputra M.**, **AK R.**, **Sharma A.**, **Sahoo S.**, **Diddee H.**, **J M.**, **Kakwani D.**, **Kumar N.**, **Pradeep A.**, **Deepak K.**, **Raghavan V.**, **Kunchukuttan A.**, **Kumar P. and Khapra M.S.** (2021). Samanantar: the largest publicly available parallel corpora collection for 11 indic languages.

Song H., **Dabre R.**, **Mao Z.**, **Cheng F.**, **Kurohashi S. and Sumita E.** (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*. Association for Computational Linguistics, pp. 279–285, Online.

Suleiman D. and Awajan A. (2022). Multilayer encoder and single-layer decoder for abstractive arabic text summarization. *Knowledge-Based Systems* **237**, 107791.

Üstün A., **Bisazza A.**, **Bouma G. and van Noord G.** (2020). UDapter: language adaptation for truly universal dependency parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, pp. 2302–2315, Online.

Vulić I., **Glavaš G.**, **Reichart R. and Korhonen A.** (2019). Do we really need fully unsupervised cross-lingual embeddings? 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)*, Hong Kong, China. Association for Computational Linguistics, pp. 4407–4418.

Wang X., **Pham H.**, **Arthur P. and Neubig G.** (2019). Multilingual neural machine translation with soft decoupled encoding. In *International Conference on Learning Representations*.

Wang Z., **K K.**, **Mayhew S. and Roth D.** (2020). Extending multilingual BERT to low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics, pp. 2649–2656, Online.

Wang Z., **Xie J.**, **Xu R.**, **Yang Y.**, **Neubig G. and Carbonell J.G.** (2020). Cross-lingual alignment vs joint training: a comparative study and a simple unified framework. In *International Conference on Learning Representations*.

Woller L., **Hangya V. and Fraser A.** (2021). Do not neglect related languages: the case of low-resource Occitan cross-lingual word embeddings. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, Punta Cana, Dominican Republic. Association for Computational Linguistics, pp. 41–50.

Wu S. and Dredze M. (2020). Do explicit alignments robustly improve multilingual encoders? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, pp. 4471–4482, Online.

https://doi.org/10.1017/nlp.2024.13 Published online by Cambridge University Press

Yadav S., **Gupta D.**, **Abacha A.B. and Demner-Fushman D.** (2022). Question-aware transformer models for consumer health question summarization. *Journal of Biomedical Informatics* **128**, 104040.

Zhang C., **Lai Y.**, **Feng Y. and Zhao D.** (2021). A review of deep learning in question answering over knowledge bases. *AI Open* **2**, 205–215.

Zhang M., **Liu Y.**, **Luan H. and Sun M.** (2017). Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada. Association for Computational Linguistics, pp. 1959–1970.

Zhu C. (2021a). Chapter 6 - pretrained language models. In Zhu C. (ed), *Machine Reading Comprehension*. Elsevier, pp. 113–133.

Zhu C. (2021b). Chapter 8 - applications and future of machine reading comprehension. In Zhu C. (ed), *Machine Reading Comprehension*. Elsevier, pp. 185–207.

Cite this article: Pandya HA and Bhatt BS. Does learning from language family help? A case study on a low-resource question-answering task. *Natural Language Processing* https://doi.org/10.1017/nlp.2024.13