IndicSQuAD : A Comprehensive Multilingual Question Answering Dataset for Indic Languages

Sharvi Endait, Ruturaj Ghatage, Aditya Kulkarni, Rajlaxmi Patil, Raviraj Joshi

Proceedings of the 39th Pacific Asia Conference on
Language, Information and Computation (PACLIC 39)

Emmanuele Chersoni, Jong-Bok Kim (eds.)

2025

IndicSQuAD : A Comprehensive Multilingual Question Answering Dataset for Indic Languages

Sharvi Endait[1,3], Ruturaj Ghatage[1,3], Aditya Kulkarni[1,3], Rajlaxmi Patil[1,3], Raviraj Joshi[2,3],
[1]Pune Institute of Computer Technology, [2]Indian Institute of Technology Madras, Chennai,
[3]L3Cube Labs, Pune,

Abstract

The rapid progress in question-answering (QA) systems has predominantly benefited high-resource languages, leaving Indic languages largely underrepresented despite their vast native speaker base. In this paper, we present IndicSQuAD, a comprehensive multi-lingual extractive QA dataset covering nine major Indic languages, systematically derived from the SQuAD dataset. Building on prior methods for constructing QA datasets in low-resource Indic languages, we adapt and extend translation techniques to ensure high linguistic fidelity and accurate answer-span alignment across diverse languages. IndicSQuAD comprises extensive training, validation, and test sets for each language, providing a robust foundation for model development. We evaluate baseline performances using language-specific monolingual BERT models and the multilingual MuRIL-BERT. The results indicate some challenges inherent in low-resource settings. Moreover, our experiments suggest potential directions for future work, including expanding to additional languages, developing domain-specific datasets, and incorporating multimodal data.

1 Introduction

Question Answering (QA) has been a cornerstone of natural language understanding (NLU), with datasets like SQuAD (Rajpurkar et al., 2016) driving advancements in machine learning models for extractive QA. While these datasets have enabled significant progress, most large-scale QA datasets are centered around English and a few high-resource languages, leaving many Indic languages underrepresented. Recent benchmarks highlight this disparity, with evaluations showing that multilingual Large Language Models (LLMs) perform substantially worse on low-resource languages compared to English. As a result, developing robust QA systems for Indic languages remains a challenge, despite the large native speaker population and the growing need for AI applications across diverse linguistic landscapes.

Indic languages, including Hindi, Bengali, Tamil, Telugu, Marathi, and others, are spoken by over a billion people. However, unlike English or Chinese, Indic languages lack extensive QA datasets, limiting the performance and adaptability of multilingual models in real-world applications. While initiatives such as TyDiQA (Clark et al., 2020) and XQuAD (Artetxe et al., 2019) have attempted to introduce non-English QA datasets, these remain either limited in size or do not comprehensively cover multiple Indic languages. The Indic-QA Benchmark (Singh et al., 2025) introduced in 2024 represents a significant advancement in this space, covering 11 major Indian languages from two language families and incorporating both extractive and abstractive question-answering tasks. This benchmark combines existing datasets with English QA datasets translated into Indian languages, demonstrating the growing recognition of the need for comprehensive multilingual QA resources. Nevertheless, the performance of multilingual models on this benchmark remains subpar, particularly for low-resource languages, underscoring the persistent challenges in this domain. A more targeted effort in this direction is the IndicQuest benchmark (Rohera et al., 2024), which is specifically designed to evaluate the factual accuracy of Indic LLMs.

The absence of large-scale, high-quality annotated QA datasets for Indic languages restricts their integration into information retrieval, education, healthcare services, governance applications, and AI-powered customer support systems. This limitation perpetuates digital inequality, where speakers of high-resource languages benefit more from technological advancements than those speaking low-resource languages.

To bridge this gap, we introduce IndicSQuAD, a comprehensive multi-lingual QA dataset covering

9 Indic languages, built by systematically translating and adapting the English SQuAD dataset while ensuring linguistic accuracy and answer-span alignment. Following earlier work on developing Marathi QA datasets, this study extends the methodology to nine additional Indic languages. Drawing from recent advances in alignment techniques and span retrieval methods, our approach addresses the challenges identified in previous translation efforts (e.g., morphological variations, syntactic differences, and maintaining contextual integrity). IndicSQuAD represents the largest multi-Indic QA resource to date, presenting the dataset along with additional baseline models, designed to facilitate research in low-resource language modeling and improve access to knowledge for Indic language speakers. The key contributions are as follows:

1. **Creation of IndicSQuAD dataset**[1] – A large-scale multi-lingual extractive QA dataset for 10 Indic languages, derived from SQuAD, ensuring high linguistic fidelity and comprehensive coverage across language families. The languages supported are Marathi, Hindi, Bengali, Telugu, Tamil, Gujarati, Punjabi, Kannada, Oriya, and Malayalam.

2. **Comprehensive Baseline Models and Benchmarking** – Strong baseline performances are established using fine-tuned monolingual BERT models for each Indic language, ensuring a tailored evaluation that captures language-specific nuances. Additionally, a comparative analysis with multilingual models like MuRIL Bert (Khanuja et al., 2021) is provided, assessing their effectiveness across diverse Indic languages. This evaluation framework addresses the unique challenges of each language, enabling meaningful comparisons across language families and resource availability.

2 Related Work

The development of question-answering systems for Indian languages has gained significant attention in recent years, though these languages remain resource-scarce compared to English. Several datasets have been created to address this gap using translation and native language approaches. This section provides a comprehensive overview of the existing work in this domain.

2.1 Translation Based Approaches

MahaSQuAD (Ruturaj et al., 2023) (Joshi, 2022b) represents the first comprehensive question-answering dataset specifically developed for Marathi. This work filled a critical gap in the language resources landscape. The paper details that MahaSQuAD consists of 118,516 training, 11,873 validation, and 11,803 test samples, accompanied by a gold test set of 500 manually verified examples. The work also presents a generic approach for translating SQuAD into any low-resource language, addressing the significant challenge of mapping answer translations to their spans in translated passages. In the current work, this approach is extended to nine more Indian languages.

(Kumar et al., 2022a) developed an extensive resource with 28,000 samples each for Hindi and Marathi by translating SQuAD 2.0, helping address data scarcity for these languages.

XQuAD (Artetxe et al., 2019) consists of 240 paragraphs and 1,190 question-answer pairs derived from SQuAD v1.1 and professionally translated into ten languages, including Hindi. This dataset has served as an important benchmark for cross-lingual question answering evaluation.

MLQA (Lewis et al., 2019) serves as another important benchmark with 4,918 context-question-answer triples available in Hindi. It enables the evaluation of cross-lingual generalization capabilities in multiple languages simultaneously.

2.2 Natively annotated datasets

The ChaII Dataset (Thirumala and Ferracane, 2022) features context-question-answer triples in Hindi and Tamil gathered directly without translation. Created by native speaker annotators, this dataset presents realistic information-seeking tasks focused on Wikipedia articles. The dataset includes 1,104 questions with the Hindi portion translated into ten other Indian languages.

Recent work by (Thirumala and Ferracane, 2022) has investigated the application of transformer models pre-trained on multiple languages, specifically focusing on Hindi and Tamil question-answering, demonstrating enhanced performance in extractive QA tasks.

Additionally, the Extended Chaii dataset has been developed containing Tamil translations from the SQuAD dataset, designed specifically for

[1] https://github.com/l3cube-pune/indic-nlp/tree/main/L3Cube-IndicSQUAD

question-answering tasks in low-resource Indic languages. The dataset consists of 2,855 training instances, 460 validation instances, and 250 test instances, making it a valuable resource for Tamil language processing.

The MMQA dataset (Gupta et al., 2018) contains 5,495 question-answer pairs in English and Hindi, covering factoid and short descriptive questions across multiple domains. This dataset is specifically designed to evaluate both bilingual and cross-lingual question answering that processes queries in either Hindi or English and retrieves answers in either language from documents in Hindi or English. The MMQA framework represents an important contribution toward multilingual information access, particularly beneficial in the Indian context.

2.3 Natively Annotated Datasets

While translation-based approaches have been instrumental in creating resources for low-resource languages, natively constructed datasets offer unique advantages in preserving linguistic authenticity.

BanglaQuAD (Rony et al., 2024) represents a significant contribution to Bengali language processing, containing 30,808 question-answer pairs constructed directly from Bengali Wikipedia articles by native speakers. Unlike translation-based approaches, this methodology avoids potential pitfalls associated with translated datasets, including loss of linguistic authenticity and contextual accuracy. The authors provide a detailed analysis of question types and answer distributions, along with baseline performance metrics using both monolingual and multilingual models.

The INDIC QA Benchmark (Singh et al., 2025) represents one of the most recent and comprehensive efforts in this domain, covering 11 major Indian languages and addressing both extractive and abstractive QA tasks. This benchmark aims to standardize evaluation across multiple Indic languages, enabling more direct comparisons of model performance across linguistic boundaries. The benchmark incorporates various question types and difficulty levels, providing a nuanced understanding of model capabilities across different linguistic structures found in Indic languages.

L3Cube-IndicQuest (Rohera et al., 2024) takes a more comprehensive approach by covering 19 Indic languages, making it one of the most linguistically diverse QA datasets available. Unlike many existing datasets that focus primarily on question-answering capabilities in general domains, L3Cube-IndicQuest specifically addresses five domains of particular relevance to the Indian context: Literature, History, Geography, Politics, and Economics. Each language subset contains carefully curated question-answer pairs designed to evaluate a model's ability to represent and process knowledge specific to Indian cultural and regional contexts.

The ChAII Dataset (Singh et al., 2025) features context-question-answer triples in Hindi and Tamil gathered directly without translation. Created by native speaker annotators, this dataset presents realistic information-seeking tasks focused on Wikipedia articles. The dataset includes 1,104 questions with the Hindi portion translated into ten other Indian languages.

3 Experimental Setup

3.1 Data Collection

The data collection process utilized the Stanford Question Answering Dataset (SQuAD 2.0), originally in English, which comprises over 150,000 question-answer pairs. Notably, about 34% of these questions are unanswerable, challenging models to handle ambiguity and non-definitive answers effectively. Each row in SQuAD 2.0 includes essential components such as title, context, question, answer start index, and answer text.

To create datasets, the robust translation and transliteration procedure involved developing a sophisticated algorithm to address inconsistencies that arise during translation, particularly in locating the correct answer index post-translation. By doing so, it was ensured that the translated datasets accurately reflect the nuances of the original English dataset while adapting to the linguistic characteristics of the target language. This approach not only enhances the quality of the translated datasets but also facilitates the development of more accurate question-answering models for low-resource languages.

3.1.1 Translation Methodologies

When creating multilingual QA datasets through translation, several methodologies can be employed, each with its advantages and challenges. Beyond the approach based on MahaSQuAD (Ruturaj et al., 2023), other researchers have explored various techniques for cross-lingual transfer in QA contexts.

Language	Model	EM%	F1%	EM (Has_ans)	F1 (Has_ans)	EM (No_ans)	F1 (No_ans)	BLEU% (Unigram)	BLEU% (Bigram)
Hindi	HindiRoBERTa	56.20	59.67	50.79	57.8	61.50	61.50	61.8	53.5
	MurilBERT	53.21	56.89	53.05	60.49	53.37	53.37	62.8	55.0
Punjabi	PunjabiBERT	51.04	54.53	47.59	54.63	54.42	54.42	54.4	46.1
	MurilBERT	50.80	54.41	47.08	54.38	54.40	54.40	54.5	46.3
Gujarati	GujaratiBERT	49.00	52.91	47.36	55.26	50.61	50.61	52.8	44.2
	MurilBERT	48.09	52.24	47.63	56.00	48.54	48.54	54.7	47.0
Kannada	KannadaBERT	50.97	54.90	48.54	56.49	53.34	53.34	52.3	45.9
	MurilBERT	49.64	53.81	48.27	56.68	50.99	50.99	53.7	44.6
Tamil	TamilBERT	50.97	54.44	46.38	53.39	55.47	55.47	53.03	44.26
	MurilBERT	49.70	53.14	46.40	53.34	52.94	52.94	53.85	44.83
Bengali	BengaliBERT	50.07	54.27	46.93	55.42	53.14	53.14	57.7	49.5
	MurilBERT	49.36	53.70	46.98	55.75	51.70	51.70	56.6	48.2
Telugu	TeluguBERT	52.17	55.34	44.98	51.37	59.24	59.24	54.8	47.5
	MurilBERT	51.11	54.38	44.30	50.90	57.82	57.82	52.9	45.1
Oriya	OdiaBERT	54.33	57.60	44.61	51.22	63.82	63.82	56.8	48.6
	MurilBERT	48.65	52.47	43.75	51.48	53.44	53.44	49.7	41.7
Malayalam	MalayalamBERT	51.02	49.42	42.24	49.26	59.59	59.59	52.0	43.2
	MurilBERT	45.67	49.62	41.89	49.89	49.36	49.36	46.9	38.6
Marathi	MahaBERT	51.28	54.88	51.04	58.31	51.52	51.52	57.9	49.9
	MurilBERT	50.13	53.91	51.26	58.92	49.03	49.03	57.7	49.4

Table 1: Performance of various models on different languages.

(Kumar et al., 2022b) proposed Multilingual Contrastive Training (MuCoT), a three-stage pipeline for question-answering in low-resource languages. This approach utilizes translation and transliteration with contrastive training across language families, showing particular effectiveness when data from the same language family is grouped. Their experiments demonstrated that translations from Indo-Aryan languages (Bengali and Marathi) significantly improved performance on Hindi, while Dravidian language data (Telugu and Malayalam) enhanced Tamil performance.

More recently, Self-Translate-Train (Ri et al., 2024) has emerged as a promising approach that leverages large language models to generate translations without requiring external translation systems. This method generates synthetic training data in the target language by utilizing the model's own translation capabilities, demonstrating substantial performance gains across several non-English languages without intensive additional data collection.

In creating IndicSQuAD, these approaches were built upon while addressing the specific challenges of Indic languages, such as maintaining context and handling linguistic nuances during translation. The methodology focused particularly on ensuring accurate mapping of answer spans in translated passages, a significant challenge when dealing with languages that differ substantially in word order and sentence structure from English.

Language	**Family**	**Script**
Marathi	Indo-Aryan	Devanagari
Hindi	Indo-Aryan	Devanagari
Punjabi	Indo-Aryan	Gurmukhi
Bengali	Indo-Aryan	Bengali
Gujarati	Indo-Aryan	Gujarati
Oriya	Indo-Aryan	Oriya
Tamil	Dravidian	Tamil
Telugu	Dravidian	Telugu
Kannada	Dravidian	Kannada
Malayalam	Dravidian	Malayalam

Table 2: Languages, their Families, and Scripts

3.2 Languages covered

IndicSQuAD includes question-answering datasets for 9 Indic languages, covering a diverse set of Indo-Aryan and Dravidian languages. These languages vary significantly in terms of script, morphology, and linguistic resources, making the dataset a valuable resource for multilingual and low-resource NLP research.

India's linguistic diversity is immense, with the 2011 Census identifying 122 major languages and 1,599 other languages. Among these, the most widely spoken languages are Hindi, Bengali, Marathi, Telugu, Tamil, Gujarati, Punjabi, Kannada, Odia, and Malayalam. Despite their extensive use, many of these languages are considered low-resource in the field of Natural Language Processing (NLP) due to the limited availability of annotated datasets and linguistic tools. This scarcity poses significant challenges in developing robust NLP applications, as models trained on high-resource languages often fail to generalize effectively to low-resource contexts. By creating comprehensive question-answering datasets for these languages, IndicSQuAD aims to bridge this gap, facilitating the development of more inclusive and effective NLP applications.

3.3 Robust approach

The creation of IndicSQuAD employed a robust translation strategy to preserve linguistic accuracy and contextual integrity in low-resource languages. To address the challenge of aligning translated answers with their corresponding spans in translated passages, the English context was first segmented into sentences. Each sentence and its associated answer were then translated into the target language. Using similarity analysis tools, the most contextually appropriate span in the translated passage was identified to match the translated answer. This approach ensured precise alignment, overcoming the common mismatch between independently translated answers and contexts.

Figure 1 illustrates the methodology employed to accurately map translated answers to their corresponding spans within translated passages. The robust algorithm developed for MahaSQuAD ensures precise alignment of translated answers within their contexts through the following steps:

1. **Sentence Segmentation**: The English context is divided into individual sentences using the NLTK library.

2. **Answer Sentence Identification**: The English sentence containing the answer is identified from the individual sentences.

3. **Translation**: Both the identified sentence and the answer are translated into Marathi (target language) using Google Translate.

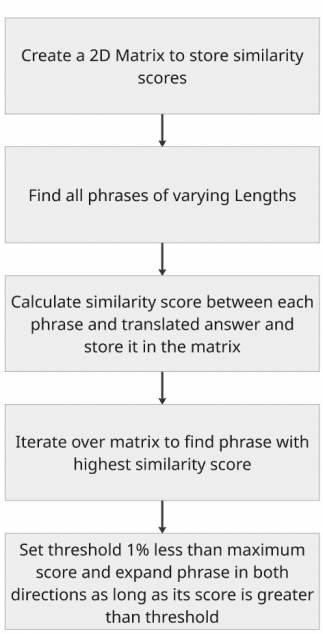

Figure 1: Algorithm for obtaining the answer and the answer span from the context

4. **Similarity Analysis**: Within the translated Marathi sentence, all possible substrings are compared to the translated answer using the SimilarityAnalyzer from the MahaNLP library (Joshi, 2022c; Magdum et al., 2023). The library uses embedding models released in (Deode et al., 2023). A similarity score matrix is generated to identify the substring with the highest similarity to the translated answer.

5. **Answer Span Determination**: The substring with the maximum similarity is selected as the base answer. Adjacent words are appended to this base answer, and the similarity is recalculated. If the new similarity score remains within 1% of the maximum, the extended phrase is accepted. This iterative process ensures that the translated answer accurately reflects the original meaning and context.

6. **Transliteration**: To maintain script consistency, named entities and numerical values are transliterated into the Devanagari script using the AI4Bharat Transliteration Engine.

To further enhance consistency, named entities were transliterated into Devanagari script using the AI4Bharat Transliteration Engine, and numerical values were converted into their counterparts. This meticulous process minimized errors and ensured uniformity across the dataset. The resulting dataset,

comprising 118,516 training samples, 11,873 validation samples, and 11,803 test samples, provides a scalable framework for translating SQuAD into other low-resource languages while maintaining linguistic and cultural nuances.

3.4 Dataset Statistics

Each of the ten languages in IndicSQuAD, including Marathi, Hindi, Bengali, Telugu, Tamil, Gujarati, Punjabi, Kannada, Oriya, and Malayalam, consists of **118,516 entities** in the training set, **11,873 entities** in the validation set, and **11,803 entities** in the test set. This large-scale dataset provides a robust foundation for training and evaluating QA models, addressing the scarcity of high-quality annotated resources for Indic low-resource languages.

Training set	118,516 samples
Validation set	11,873 samples
Test set	11,803 samples

Table 3: Dataset Statistics

4 Benchmarking and Experiments

4.1 Models used

To evaluate IndicSQuAD, monolingual and multilingual models were employed to establish baseline performances across the ten Indic languages.

- **Monolingual Models**
  Monolingual models are language-specific models trained solely on a particular language. These models are optimized for the linguistic characteristics of their respective languages, often leading to improved performance compared to multilingual counterparts. For each language in IndicSQuAD, BERT-based monolingual models for low-resource languages such as HindiBERT (Joshi, 2022a), PunjabiBERT, GujaratiBERT, KannadaBERT, TamilBERT, BengaliBERT, OdiaBERT, and MalayalamBERT were utilized, fine-tuned on the corresponding datasets.

- **Multilingual Models**
  In this research, MuRIL-BERT (Multilingual Representations for Indian Languages) was utilized, a transformer-based language model pre-trained on 17 Indian languages, including Marathi. MuRIL-BERT has demonstrated robust performance in understanding and processing Indian languages, making it a suitable choice for this study. This model's architecture allows it to effectively capture linguistic nuances across multiple Indian languages, facilitating the development of more accurate and efficient natural language processing applications.

4.2 Experimental Setup

Fine-tuning was conducted on the models using a custom dataset spanning three epochs and utilizing A100 GPUs with a consistent batch size of 32. The carefully selected hyperparameters include n_best_size (which refers to the number of predictions provided per question) set to 2, which significantly shaped the training dynamics and influenced the experimental outcomes. The other key hyperparameters employed during fine-tuning included a learning rate of 1e-4 and the AdamW optimizer. These adjustments were crucial in refining the model and enhancing its performance.

4.3 Results

The evaluation of the IndicSQuAD dataset from Table 1 highlights the superior performance of monolingual models over the multilingual MurilBERT across most Indic languages. For example, HindiRoBERTa outperformed MurilBERT for Hindi, achieving higher EM and F1 scores (56.20% and 59.67%, respectively, compared to 53.21% and 56.89%). Similarly, language-specific models like BengaliBERT and TamilBERT demonstrated better results in their respective languages, with BengaliBERT achieving an EM score of 50.07% compared to MurilBERT's 49.36%. These monolingual models consistently showed better contextual understanding and exact match accuracy.

In contrast, MurilBERT exhibited more generalized performance but lagged in capturing language-specific nuances, especially in low-resource languages like Telugu and Malayalam. For instance, TeluguBERT achieved an EM score of 52.17%, outperforming MurilBERT's 51.11% by leveraging its tailored design for Telugu. This trend underscores the importance of monolingual models in improving language-specific performance and highlights the need for further optimization of multilingual models to close the gap in low-resource language processing.

5 Conclusion and Future work

IndicSQuAD represents a significant advancement in addressing the scarcity of high-quality, large-scale question answering datasets for Indic languages. By systematically translating and adapting the widely-used SQuAD dataset into nine major Indic languages, this work not only bridges the resource gap but also establishes robust baselines using both language-specific and multilingual models. The comprehensive evaluation framework highlights the superior performance of monolingual models in capturing linguistic nuances, while also underscoring the challenges faced by multilingual models in low-resource settings. The dataset, along with the accompanying models and evaluation tools, will be made publicly available upon publication, fostering further research and development in multilingual NLP. Moving forward, expanding IndicSQuAD to additional languages, creating domain-specific datasets, and integrating multimodal data will further enhance the accessibility and effectiveness of AI-powered applications for Indic language speakers. This initiative is a crucial step toward reducing digital inequality and ensuring that speakers of low-resource languages can fully benefit from advances in natural language understanding and information retrieval.

While IndicSQuAD provides a strong foundation for question-answering (QA) tasks in Indic languages, there are several directions for future research and development:

1. **Expansion to More Languages**
   Extending the dataset to cover additional low-resource Indic languages, such as Assamese, Manipuri, and Santali, to improve multilingual accessibility and representation.

2. **Domain-Specific QA Datasets**
   Creating specialized datasets for legal, medical, and financial domains to improve real-world applicability in Indic languages.

3. **Multimodal QA for Indic Languages**
   Extending the dataset to incorporate images, videos, and speech, enabling multimodal question-answering for a more inclusive AI ecosystem.

4. **Interactive and Real-World Applications**
   Deploying QA models trained on IndicSQuAD into real-world applications, such as chatbots, voice assistants, and educational tools, to enhance accessibility and usability.

Limitations

A limitation of IndicSQuAD is that it is created through translation of SQuAD rather than native annotation. This reliance on translated data may reduce the linguistic authenticity of the contexts and questions. It can also introduce artifacts such as unnatural phrasing, loss of cultural nuances, or inconsistencies in answer-span alignment. Furthermore, the dataset may not fully reflect the diversity of information-seeking behavior found in native speakers of Indic languages.

References

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2019. On the cross-lingual transferability of monolingual representations. *CoRR*, abs/1910.11856.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. Tydi qa: A benchmark for information-seeking question answering in typologically diverse languages. *Preprint*, arXiv:2003.05002.

Samruddhi Deode, Janhavi Gadre, Aditi Kajale, Ananya Joshi, and Raviraj Joshi. 2023. L3cube-indicsbert: A simple approach for learning cross-lingual sentence representations using multilingual bert. In *Proceedings of the 37th Pacific Asia Conference on Language, Information and Computation*, pages 154–163.

Deepak Gupta, Surabhi Kumari, Asif Ekbal, and Pushpak Bhattacharyya. 2018. MMQA: A Multi-domain Multi-lingual Question-Answering Framework for English and Hindi. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Raviraj Joshi. 2022a. L3cube-hindbert and devbert: Pre-trained bert transformer models for devanagari based hindi and marathi languages. *arXiv preprint arXiv:2211.11418*.

Raviraj Joshi. 2022b. L3cube-mahacorpus and mahabert: Marathi monolingual corpus, marathi bert language models, and resources. In *Proceedings of the WILDRE-6 Workshop within the 13th Language Resources and Evaluation Conference*, pages 97–101.

Raviraj Joshi. 2022c. L3cube-mahanlp: Marathi natural language processing datasets, models, and library. *arXiv preprint arXiv:2205.14728*.

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. *Preprint*, arXiv:2103.10730.

Gokul Karthik Kumar, Abhishek Gehlot, Sahal Shaji Mullappilly, and Karthik Nandakumar. 2022a. Mucot: Multilingual contrastive training for question-answering in low-resource languages. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*, pages 15–24.

Gokul Karthik Kumar, Abhishek Singh Gehlot, Sahal Shaji Mullappilly, and Karthik Nandakumar. 2022b. Mucot: Multilingual contrastive training for question-answering in low-resource languages. *Preprint*, arXiv:2204.05814.

Patrick S. H. Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2019. MLQA: evaluating cross-lingual extractive question answering. *CoRR*, abs/1910.07475.

Vidula Magdum, Omkar Jayant Dhekane, Sharayu Sandeep Hiwarkhedkar, Saloni Sunil Mittal, and Raviraj Joshi. 2023. mahanlp: A marathi natural language processing library. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 34–40.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

Ryokan Ri, Shun Kiyono, and Sho Takase. 2024. Self-translate-train: Enhancing cross-lingual transfer of large language models via inherent capability. *Preprint*, arXiv:2407.00454.

Pritika Rohera, Chaitrali Ginimav, Akanksha Salunke, Gayatri Sawant, and Raviraj Joshi. 2024. L3cube-indicquest: A benchmark question answering dataset for evaluating knowledge of llms in indic context. *arXiv preprint arXiv:2409.08706*.

Md Rashad Al Hasan Rony, Sudipto Kumar Shaha, Rakib Al Hasan, Sumon Kanti Dey, Amzad Hossain Rafi, Amzad Hossain Rafi, Ashraf Hasan Sirajee, and Jens Lehmann. 2024. Banglaquad: A bengali open-domain question answering dataset. *Preprint*, arXiv:2410.10229.

Ghatage Ruturaj, Kulkarni Aditya Ashutosh, Patil Rajlaxmi, Endait Sharvi, and Joshi Raviraj. 2023. Mahasquad: Bridging linguistic divides in marathi question-answering. In *Proceedings of the 20th International Conference on Natural Language Processing (ICON)*, pages 497–505.

Abhishek Kumar Singh, Vishwajeet kumar, Rudra Murthy, Jaydeep Sen, Ashish Mittal, and Ganesh Ramakrishnan. 2025. Indic qa benchmark: A multilingual benchmark to evaluate question answering capability of llms for indic languages. *Preprint*, arXiv:2407.13522.

Adhitya Thirumala and Elisa Ferracane. 2022. Extractive question answering on queries in hindi and tamil.

A Appendix

Language	Model link
Marathi	marathi-squad-bert
Hindi	hindi-squad-bert
Bengali	bengali-squad-bert
Telugu	telugu-squad-bert
Tamil	tamil-squad-bert
Gujarati	gujarati-squad-bert
Punjabi	punjabi-squad-bert
Kannada	kannada-squad-bert
Oriya	oriya-squad-bert
Malayalam	malayalam-squad-bert

Table 4: Language-specific SQUAD BERT models on HuggingFace