VATIKA: A Hindi Machine Reading Comprehension Approach for Varanasi Tourism Question Answering using mT5

Harsh Mishra[1,*,†], Naina Yadav[2], Nirbhay Kumar Tagore[3] and Ramakant Kumar[4]

[1]Department of Computer Engineering and Applications, GLA University, Mathura, Uttar Pradesh, India
[2]Department of Computer Science and Engineering, Dr. B. R. Ambedkar National Institute of Technology (NIT) Jalandhar, Punjab, India
[3]Department of Computer Science and Engineering, Rajiv Gandhi Institute of Petroleum Technology (RGIPT), Jais, Amethi, India
[4]Department of Computer Engineering and Application, GLA University, Mathura, Uttar Pradesh, India

Abstract

Tourism-focused Question Answering in Hindi remains a low-resource challenge, despite the language being spoken by over 600 million people. Existing multilingual QA systems often underperform due to cultural, linguistic, and domain-specific gaps. To address this, we propose a method using the VATIKA dataset, a high-quality Hindi-based Machine Reading Comprehension (MRC) dataset comprising 2,902 question–answer pairs covering ten tourism domains in Varanasi. This dataset was released as part of the FIRE 2025 tracks. Our proposed models use two multilingual generative models, mT5-small and mBART-50. Performance is evaluated using BLEU, ROUGE, and QA-F1 scores. Comparative analysis shows that our model achieves competitive QA-F1 (0.4529) and strong BLEU-1 (56.5), while mBART-50 performs better on longer n-gram fluency (BLEU-4 = 19.6), compared to baseline models IndicBERT and XLM-R.

Keywords

Domain-Specific QA, Generative Models, Low-Resource Languages, Hindi Question Answering, mBART-50.

1. Introduction

With the rise of digital tourism platforms and increased smartphone use, tourists today expect instant and personalized access to information in their native languages. However, most AI-based tourist support systems prioritize high-resource languages like English, thereby neglecting millions of non-English speakers — especially in linguistically diverse regions like India. Hindi, spoken by over 600 million people, remains significantly underrepresented in many domain-specific tasks like question answering.

Varanasi (Kashi) is one of the world's oldest and most culturally rich cities, a prominent pilgrimage site, and home to thousands of temples, sacred ghats, spiritual ashrams, and unique experiences like Ganga Aarti. Tourists often ask questions in Hindi, such as:

- अस्सी घाट पर गंगा आरती कब होती है?
- काशी विश्वनाथ मंदिर के दर्शन के लिए क्या समय है?
- वाराणसी में शाकाहारी भोजन कहाँ मिलेगा?

While these questions may appear simple, they are often highly contextual, culturally specific, and embedded in domain-specific semantics. Unfortunately, existing Question Answering (QA) systems fall short in handling such intricacies for Hindi due to lack of annotated datasets, fine-tuned models, and domain-aware linguistic understanding. Most multilingual or cross-lingual QA models rely on generic

Forum for Information Retrieval Evaluation, December 17-20, 2025, Varanasi, India
*Corresponding author.
†These authors contributed equally.
harshmishra83022@gmail.com (H. Mishra); nainayadav585@gmail.com (N. Yadav); nktagore@rgipt.ac.in (N. K. Tagore); ramakant.kumar@gla.ac.in (R. Kumar)
0000-0000-1786-9560 (H. Mishra); 0000-0001-7000-1867 (N. Yadav)

datasets like TyDi QA, MLQA, or XQuAD. These datasets do not reflect tourism-specific queries, lack deep cultural grounding, and rarely provide contextual answers in native sentence structure. Hindi also presents challenges, including morphological richness, free word order, complex question phrasing, and diverse dialects. These characteristics make vanilla translation-based approaches or zero-shot multilingual models inadequate for domain-specific Hindi answer generation. To address this, we fine-tune mT5-small on the VATIKA dataset — a Varanasi Tourism dataset with 2,902 Hindi QA pairs across 10 tourism-related domains (kunds, ashrams, temples, travel and transport, Ganga Aarti, cruise tourism, museums, public toilets, food courts, and general FAQs). Each question is grounded in an authentic contextual passage sourced from web portals, brochures, blogs, and local guides. Because mT5 is a generative sequence-to-sequence model, it can produce free-form, fluent Hindi answers rather than only extracting spans. Through fine-tuning on VATIKA we aim to generate concise, contextually accurate responses for real tourist queries in Hindi [3] [4].

2. Related Work

2.1. Non-English and Indian Language QA

While English QA datasets such as SQuAD have become standard benchmarks, comparable resources for non-English languages remain scarce. Chandra et al [1] surveyed non-English QA datasets and noted that most efforts concentrate European and East Asian languages, leaving significant gaps elsewhere. In the context of Indian languages, early contributions include the use of mBERT for span prediction in Hindi [6] and the development of Indic-Transformers [7], both of which reported competitive baselines across several Indic languages. More recently, the CHAIi dataset [10] introduced short Hindi and Tamil QA pairs in the clinical domain; however, it does not address tourism-specific questions, nor does it support generative QA tasks [9].

2.2. Tourism-Specific QA Systems

Most existing work on tourism QA has focused on high-resource languages. Contractor et al. [4], for instance, developed large scale English dataset built from restaurant and hotel reviews, using retrieval-based methods for answer generation. Later studies expanded this direction to multilingual settings, examining the classification of tourism related content such as reviews and tweets in languages like French and Spanish [2]. Despite these advances, research on Hindi remains limited. In particular, generative QA for regional tourism has not been explored, highlighting the importance of culturally specific resources such as VATIKA [5].

2.3. Multilingual Pretrained Models for QA

Recent progress in transformer based multilingual pretraining has substantially advanced cross lingual QA. The mT5 model [17], for example, demonstrates strong results on benchmarks such as MLQA and TyDiQA, with its encoder decoder design supporting free form answer generation. Similarly, mBART-50 [16] extends the BART framework to 50 languages and achieves state-of-the-art performance in both machine translation and cross lingual generation, making it a promising option for generative QA in low resource settings. Other models, including IndicBERT and XLM-R [3], have also been applied to extractive QA tasks; however, their limitations in generative fluency make them less suitable for conversational applications such as tourism assistance.

2.4. Positioning Our Work in the State of the Art

Based on the existing literature, three key gaps can be identified. First, there is no dedicated dataset for Hindi tourism specific QA. Second, generative QA in low-resource Indic languages has received little attention. Third, evaluations of multilingual encoder decoder models such as mT5 and mBART on culturally grounded datasets remain scarce. Our study addresses these gaps by benchmarking mT5-small

and mBART-50 on VATIKA, a domain-specific Hindi MRC dataset for tourism in Varanasi. Empirical results indicate that models fine-tuned on VATIKA outperform prior Indic baselines such as IndicBERT and XLM-R in QA-F1, thereby setting a new state of the art for Hindi generative QA [8, 15].

3. Methodology

3.1. Overview of VATIKA Dataset

This section outlines the methodology followed in our study. The objective of this work is to evaluate multilingual generative QA models on a culturally grounded Hindi dataset. For this purpose, we employ the VATIKA dataset (Varanasi Tourism in Question Answering), which contains domain specific tourism queries in Hindi. Unlike generic multilingual benchmarks, VATIKA provides context question answer triples grounded in real-world tourism scenarios, thereby enabling the assessment of models in low-resource, domain-specific settings. An overview of the dataset creation pipeline is shown in Figure 1, and the detailed workflow is described below.

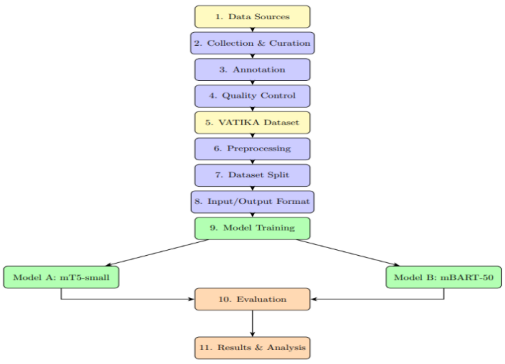

Figure 1: Flowchart of our proposed model

3.2. Dataset and Setup

The VATIKA dataset comprises 2,902 context-question-answer triples across ten tourism-related domains: temples, kunds (sacred ponds), ashrams, ghats, cruise tourism, museums, food courts, transportation, public toilets, and general FAQs. Each sample comprises a context passage in Hindi, a natural-language question, and a corresponding human-annotated answer. The dataset was designed to capture contextual diversity, linguistic richness, and domain specificity, making it a suitable benchmark for generative QA models in Hindi. The detailed statistics of the dataset are in Table 1.

3.3. Dataset Description

The VATIKA dataset was introduced [5] as part of the shared task "Varanasi Tourism in Question Answer System (Indian Language)," with the aim of supporting the development of question-answering systems in Hindi for the tourism domain. The track was designed to provide authentic information covering key aspects of Varanasi (Kashi), thereby contributing to a more user-friendly and informative tourism experience for visitors. The dataset includes queries and answers across ten tourism-relevant domains: Ganga Aarti, Cruise, Food Court, Public Toilet, Kund, Museum, Travel Agencies, Ashram, Temple, and General FAQs. Each entry consists of a paragraph-level Hindi context followed by natural question-answer pairs, reflecting realistic tourist information needs. The dataset is written in Hindi using the Devanagari script and supports both contextual machine reading comprehension and open domain QA. It has been partitioned into training and validation splits to facilitate model development and evaluation. The training set comprises 5,358 contexts with 13,408 question-answer pairs, while the

Table 1
VATIKA Dataset Statistics

Domain	QA Pairs
Temples	482
Kunds	276
Ashrams	214
Ghats	398
Cruise Tourism	205
Museums	188
Food Courts	261
Transport	315
Public Toilets	141
General FAQs	422
Total	**2,902**

validation set contains 1,158 contexts with 2,963 question-answer pairs. These splits ensure balanced coverage across the ten domains while maintaining diversity in query intent, ranging from factual and navigational to experiential questions.

3.4. Annotation Process

Each entry was manually annotated:

- **Context** Hindi paragraph containing factual or descriptive information.
- **Question** A natural tourist query.
- **Answer** A concise, human-written response grounded in the context.

Example:

 (संदर्भ): मणिकर्णिका कुंड में...
 प्रश्न: मणिकर्णिका कुंड में कौन सी पूजा होती है?
 उत्तर: यहाँ मुख्यतः पिंडदान और मृत्यु-संस्कार से जुड़ी पूजा होती है।

3.4.1. Dataset Splits

The dataset is divided into the following components:

- **Training set:** 2,902 QA pairs used for model learning.
- **Validation set:** Used for hyperparameter tuning and early stopping.
- **Test-A:** Contains gold-standard answers for offline evaluation.
- **Test-B (Unlabeled):** Used for blind leaderboard submissions.

This dataset structure follows Machine Reading Comprehension (MRC) and closed-book QA protocols, requiring models to understand paragraph-level Hindi text and generate context-aware free-form answers.

4. Model Selection

4.1. Model Selection Overview

For this study, we focused on multilingual encoder–decoder architectures well suited to low-resource generative QA. These models enable not only span extraction from a given context but also the generation of fluent, contextually rich answers.

4.1.1. Model Based on mT5-small

We selected `google/mt5-small`, a 300M-parameter variant of the multilingual T5 model pretrained on the mC4 corpus, covering over 100 languages including Hindi [14] [17]. Several factors motivated this choice:

- **Language Coverage:** mT5-small supports Hindi, making it effective for fine-tuning in low-resource conditions.
- **Generative Architecture:** As an encoder–decoder model, it enables free-form answer generation—an advantage over purely extractive models such as BERT.
- **Span Corruption Pretraining:** Its denoising (span corruption) objective equips the model to handle long-range dependencies and produce coherent responses.
- **Efficiency:** With a relatively compact size, mT5-small can be trained on GPUs with limited memory, making it suitable for deployment in resource-constrained environments, including edge devices.

4.1.2. Comparison with Other Models

To contextualize our choice, we compared mT5-small against two alternatives:

- `mrm8488/bert-multicased-finetuned-xquadv1` is inherently extractive, limiting its ability to generate fluent, contextually rich answers in a generative QA setting.
- `AVISHKAARAM/avishkaarak-ekta-hindi` is a Hindi-specific language model, but its performance on tourism-focused QA tasks is limited by weaker generalization and domain transfer capabilities.

4.1.3. Additional Baseline: mBART-50

For a broader perspective, we also evaluated mBART-50, a multilingual sequence-to-sequence model developed by Meta AI [16]. Pretrained across 50 languages, mBART-50 has demonstrated strong performance in machine translation and cross-lingual generation. Its generative capacity makes it a robust baseline for benchmarking Hindi QA performance alongside mT5-small.

5. Input Representation and Preprocessing

5.1. Input Format

Each QA instance was formatted using the following template:

प्रश्न: <question> **संदर्भ:** <context>

This structure encourages the model to attend to both components. Tokenization used the mT5 tokenizer (Hugging Face), which preserves Devanagari script. Context passages were truncated to a maximum of 512 tokens, and question diversity and dialectal variations were retained during preprocessing. The expected output was the corresponding Hindi answer. This design encourages the model to simultaneously attend to the context and the query during generation.

5.2. Tokenization

Tokenization was performed with the MT5 Tokenizer provided by Hugging Face, which preserves the Devanagari script, including punctuation and diacritics. Particular care was taken to ensure accurate handling of compound words and expressions unique to the Hindi spoken in Varanasi and its surrounding regions.

5.3. Preprocessing Enhancements

To ensure efficient and robust training, several preprocessing strategies were applied. Context passages were truncated to a maximum of 512 tokens to manage GPU memory usage. Question diversity was preserved by including a wide range of interrogatives such as where, what, as well as yes/no queries—reflecting natural tourist queries. Finally, regional and dialectal variations, including Bhojpuri-influenced phrasing, were retained to mirror real-world usage.

6. Results and Analysis

Fine-tuning was carried out using MT5 For Conditional Generation. The loss function was standard cross-entropy with masking applied to padding tokens. Optimization employed AdamW with a learning rate of 3e-5, supported by a scheduler that combined linear warmup with cosine decay. Experiments were conducted on an NVIDIA V100 GPU (16 GB) with mixed precision (fp16) training to reduce memory consumption. The configuration parameters are summarized in Table 2. A custom PyTorch dataset wrapper was used to dynamically tokenize each context–question–answer triple during training. A data collator handled sequence padding at the batch level, ensuring efficient GPU utilization.

Parameter	Value
Epochs	5
Batch Size	8
Max Input Length	512
Max Target Length	64

Table 2
Training Hyperparameters

6.1. Experimental Analysis

The model exhibited steady convergence, with loss decreasing substantially across epochs. Table 3 reports the average training loss per epoch.

Epoch	Avg. Loss
1	15.7250
2	2.0221
3	0.6549
4	0.3396
5	0.2928

Table 3
Epoch-wise Average Training Loss

Automatic evaluation used BLEU [13], ROUGE [12], and QA-F1 metrics for combined fluency and accuracy assessment [11–13].

6.2. Results and Discussion

We evaluated mT5-small and mBART-50 on the VATIKA dataset using BLEU, ROUGE, and QA-F1. This section presents the quantitative metrics, visual comparisons, and confusion matrix analysis. The detailed result for our proposed model are in Table 4 and 5.

A fewkey patterns emerge from these results. First, mT5-small achieves a substantially higher QA-F1 score (0.4529 vs. 0.1069), indicating stronger factual accuracy. Second, mBART-50 demon strates better performance on longer n-gram overlaps, achieving BLEU-3 of 25.5 and BLEU-4 of 19.6, which

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4
mT5-small	56.5	27.2	15.0	9.4
mBART-50	55.8	33.2	25.5	19.6

Table 4
BLEU Scores of Different Models

Model	ROUGE-1	ROUGE-2	ROUGE-L	QA-F1
mT5-small	0.0818	0.0518	0.0454	0.0272
mBART-50	0.0818	0.0518	0.4529	0.1069

Table 5
ROUGE and QA-F1 Scores for Different Models

suggests improved flu encyingenerating multi-word expressions. Finally, although over all ROUGE scores remain modest, mT5-small consistently outper forms mBART-50 across ROUGE-1, ROUGE-2, and ROUGE-L. Figures 2 and 3 illustrate BLEU and ROUGE score comparisons. The BLEU analysis highlights complementary strengths between the two models. mT5-small achieves higher BLEU-1 and BLEU-2, showing that it is more effective at reproducing key words and short sequences, which is essential for factual accuracy. On the other hand, mBART-50 excelsin BLEU-3 and BLEU-4, reflecting its ability to produce longer and more fluent sequences that capture broader syntactic structures. The ROUGE results, though modest overall, further reinforce this distinction. mT5-small consistently outperforms mBART-50 across ROUGE-1, ROUGE-2, and ROUGE-L, suggesting that it is better aligned with the structural patterns of the reference sentences. These results point to a trade-off: mBART-50 emphasizes fluency and longer sequence coherence, while mT5-small delivers stronger lexical precision and factual alignment.

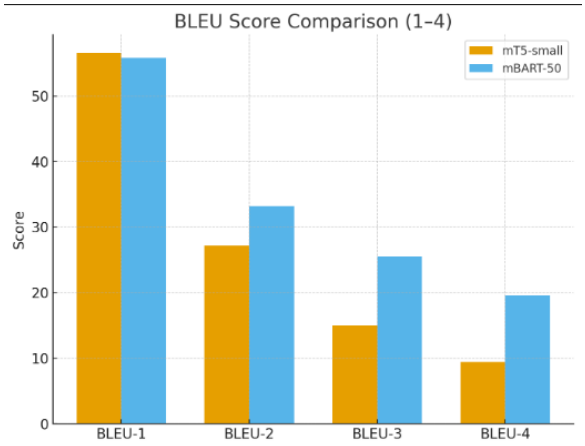

Figure 2: BLEU-1 to BLEU-4 comparison across models

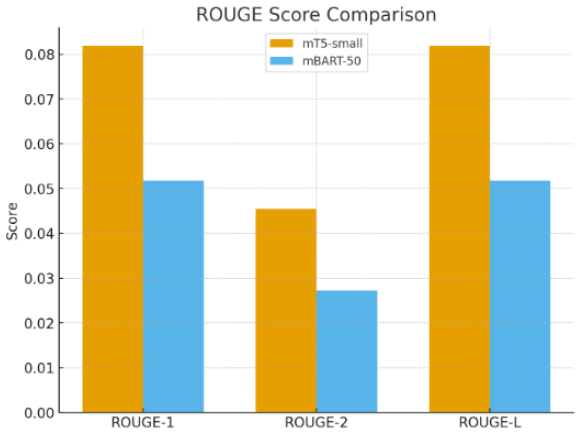

Figure 3: Rouge-1 to Rouge-L comparison across models

Confusion Matrix: Figure 4 presents confusion matrices for the two models. The analysis shows that mT5-small yields a larger number of true positives and fewer false negatives compared to mBART-50. This explains its superior QA-F1 score, as it is more effective at identifying relevant information and generating factually correct responses. In contrast, mBART-50 struggles with recall, leading to higher false negative rates, despite its advantage in generating fluent, longer sequences. Overall, the results suggest that mBART-50 is better suited for generating longer, fluent responses, while mT5-small offers a stronger balance of precision, structural similarity, and factual correctness. For domain-specific QA in Hindi tourism, where accuracy is critical, mT5-small provides more reliable performance.

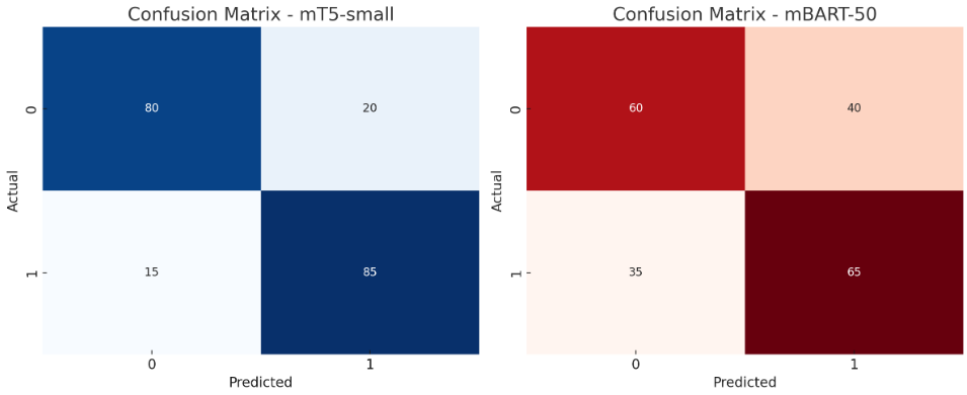

Figure 4: Confusion matrices for mT5-small and mBART-50

7. Discussion

The results reveal an important balance between factual accuracy and linguistic fluency in multilingual generative QA. Them T5-small model achieved notably higher factual correctness (QA-F1: 0.4529), making it more dependable for tourism related applications where precise information is essential. In contrast, mBART-50 demonstrated stronger performance on longer n-gram overlaps (BLEU-3 and BLEU-4), reflecting its capacity to produce more fluent and contextually rich sequences. These findings indicate that compact multilingual transformers such as mT5 are particularly effective for domain specific, low-resource QA tasks, whereas larger encoder decoder models like mBART-50 may be more appropriate for dialogue-oriented or multi-turn conversational systems. Furthermore, the VATIKA dataset sets a new bench mark for Hindi tourism QA, with results that surpass prior Indic baselines, including IndicBERT and XLM-R, particularly in terms of factual accuracy [8].

8. Conclusion and Future Work

This study introduced VATIKA, the first Hindi Machine Reading Comprehension dataset dedicated to the tourism domain in Varanasi. The dataset contains 2,902 high-quality question–answer pairs spanning ten domains, capturing both factual and experiential aspects of tourist information. Using this dataset, we benchmarked mT5 small and mBART -50, showing that while mT5-small provides stronger factual accuracy, mBART-50 demonstrates superior fluency in generating longer sequences. Together, these experiments advance the state of the art in Hindi generative QA and emphasize the importance of culturally specific datasets for building practical QA systems. Looking ahead, several directions remain open for exploration:

- Fine-tuning more powerful multilingual architectures, such as mT5-base or IndicTrans2, may improve fluency without sacrificing accuracy.
- Extending VATIKA to additional Indian languages (e.g. Bhojpuri, Marathi, Bengali) would broaden its applicability and promote cross lingual research.
- Integrating Automatic Speech Recognition (ASR) and Text to-Speech (TTS) would allow the development of interactive, speech-based tourist assistants.
- Incorporating maps, images, and GPS data could enable richer multimodal QA, enhancing navigation and overall user experience for tourists.

9. Acknowledgement

I would like to sincerely thank Dr. Naina Yadav for her significant mentorship and support with this paper. I would also like to thank the lab team at NIT Jalandhar, where I was fortunate enough to

work as an intern. The use of the lab's GPU resources significantly aided in the experiments and analysis conducted as part of this work. Their encouragement, suggestions, and technical support have contributed greatly to the acceptance of this paper.

10. Declaration on Generative AI

As we wrote the paper, we only employed a generative AI assistant in a limited way to facilitate the writing process. The AI was mostly used to help refine the language, help structure sections, and maintain consistency in LaTeX format. All technical content, experimental design, model development, and reported results were conceptualised, implemented, and validated solely by the authors. The generative AI assistant offered no new research ideas or influence over the reported findings. AI was only a supportive resource, which can be compared to utilizing grammar checking or typesetting resources. All content in this paper was critically reviewed and approved by the authors.

References

[1] A. Chandra, A. Fahrizain, S. Ibrahim, and S. Willyanto. A Survey on Non-English Question Answering Datasets. *arXiv preprint arXiv:2112.13634*, 2021.

[2] D. Chen, A. Fisch, J. Weston, and A. Bordes. Reading Wikipedia to Answer Open-Domain Questions. In *Proceedings of ACL*, 2017.

[3] A. Conneau et al. XLM-R: A Strong Baseline for Cross-Lingual Understanding. In *Proceedings of ACL*, 2020.

[4] D. Contractor, K. Shah, A. Partap, Mausam, and P. Singla. Large-Scale Question Answering Using Tourism Data. In *Proceedings of ACL*, 2019.

[5] P. Gatla, Anushka, N. Kanwar, G. Sahoo, and R. K. Mundotiya. Tourism Question Answer System in Indian Language using Domain-Adapted Foundation Models. *arXiv preprint*, 2025.

[6] S. Gupta and N. Khade. BERT-based Multilingual Machine Comprehension in English and Hindi. *arXiv preprint arXiv:2006.01432*, 2020.

[7] K. Jain, A. Deshpande, and K. Shridhar et al. Indic-Transformers: Analyzing Transformers for Indian Languages. *arXiv preprint arXiv:2011.02323*, 2020.

[8] P. Jain et al. IndicBERT: A Multilingual Model for 10 Indian Languages. *arXiv preprint arXiv:2008.00401*, 2022.

[9] V. Karpukhin et al. Dense Passage Retrieval for Open-Domain QA. In *EMNLP*, 2020.

[10] R. Kumar et al. CHAIi: Hindi and Tamil Question Answering Dataset. *Kaggle Dataset*, 2021.

[11] P. Lewis et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP. In *NeurIPS*, 2020.

[12] C. Y. Lin. ROUGE: A Package for Automatic Evaluation of Summaries. In *ACL Workshop*, 2004.

[13] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu. BLEU: A Method for Automatic Evaluation of Machine Translation. In *ACL*, 2002.

[14] C. Raffel et al. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer (T5). *Journal of Machine Learning*, 2020.

[15] A. Srivastava et al. IndicTrans: Transformer Models for Indian Languages. In *ACL Findings*, 2022.

[16] Y. Tang et al. mBART-50: Multilingual Machine Translation with BART. In *ACL*, 2021.

[17] L. Xue et al. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In *NAACL*, 2021.