Multilingual vs Indic Models for Hindi NLP Tasks

Dileep Patel(22683)
Department of Computer Science and Automation
Indian Institute of Science
dileeppatel@iisc.ac.in

Abstract

Large Language Models (LLMs) have brought a big change in handling Natural Language Processing (NLP) tasks like text classification, question answering, and summarization. But most of these LLMs are trained on global multilingual data, which makes them less effective for Indian languages like Hindi, as these languages have their own unique scripts, grammar, and challenges. This study compares the performance of generalized multilingual models like XLM-RoBERTa and mBERT with specialized Indic models like MuRIL and IndicBERT-v2 for Hindi NLP tasks, focusing on document classification and extractive question answering. My findings show that specialized Indic models perform better in classification tasks because they are trained on data specific to Indian languages. But for question answering, we noticed that generalized models sometimes give better results because they can handle translated content and broader contexts more effectively. This study shows why having specialized models for Indian languages is so important. These models can make a huge impact in improving NLP tasks, especially for low-resource languages like Hindi.

1 Introduction

Large Language Models(LLMs) has been in the heart of advancements in Natural Language Processing(NLP). Especially for tasks like, Document Classification, Question Answering, Text Summarization, Natural Language Inference, etc., the use of LLMs are inevitable nowadays. After being trained on Large Corpus of Internet Data for significant amount of time using enormous computing resources, we could see such performance in the models.

However, major LLMs are focused on English Language and its siblings which is justified because almost 52% of Internet Data is in English[1]. But, for the aim of globalization of AI technologies, it became a need for multilingual models in these tasks too. Hence, [6] came with mBert, a transformer-based model trained on 104 languages using a shared subword vocabulary. Following the same pathway, [4] extended mBERT by incorporating supervised translation tasks during training. GPT-3 [3], mT5 [15], BLOOM [14], etc. showed the potential of multilingual modeling in generative tasks too.

In parallel, the need for specialized model in the domain of Indic Languages came forward due to its different scripts and language structure. (INDICBert)[11] improved performance on IndicNLP tasks like text classification and named entity recognition. , INDICBart [5] is based on the BART (Bidirectional and Auto-Regressive Transformer) architecture but fine-tuned and optimized for tasks involving Indian languages. Similarly, MURIL[13] was a fine-tuned multilingual model optimized for Indian languages, outperforming mBERT in Indian contexts.

In this term report, I have done a comparative survey of multilingual models and Indian specialized models on two major tasks Document Classification and Extractive Question Answering. The findings suggest that specialized Indic models perform better for document classification due to their focus on Indian language structures, while generalized models excel in handling machine-translated content for question answering tasks.

[0]Code: https://github.com/dileepp982/DLNLP_Project, Data: link

2 Problem Statement

The main objective of this experiment is to explore how good the specialized Indic Language Models are with respect to general Multilingual Language Models for Hindi language based NLP tasks. This comparison is justified because specialized Indic Language Models are trained to take account of the intricacies of Devanagari and Brahmi Scripts, different Grammars and unique dialects. And, their comparison with general models would give an insights of their ability to outperform and hence prove the usefulness of their independent existence. Hence, this report highlights the comparative analysis of the performance general multilingual LLMs vs specialized Indic Language LLMs for Document Classification and Question Answering for hindi language.

For Document Classification, datasets used are BBC News Headline, BBC News Article and IIT Patna Reviews. Similarly, for extractive Question Answering, datasets used are MergedQuAD, IndicQA and ChAII. For both the tasks, the models fine-tuned are **Generalised MultiLingual Model:** XLM-RoBERTa [4] and mBERT [6]; **Specialized Indic Model:** MURIL [13] and IndicBERT [11].

In the experiment, Specialized Indic models (MuRIL, IndicBERT-v2) outperform generalized multilingual models (XLM-RoBERTa, mBERT) in Hindi document classification, while generalized models perform better in extractive question answering tasks involving machine-translated datasets.

3 Methodology

3.1 Dataset Description

[Table 1] For the task of **Document Classification**, I have taken datasets BBC News Headline, BBC News Article and IIT Patna Reviews. **BBC News Headline**[12] Originally created for Natural Language Inference (NLI), this dataset contains textual entailment pairs in Hindi. It contains four columns: Premise, Hypothesis, Label, and Topic. For this task, only the Premise (news headline) and Topic columns were used, with Topic indicating one of six predefined topics. This dataset is ideal for classifying short and focused text like news headlines. **BBC News Article**[8] This dataset contains full Hindi news articles and their corresponding categories, with two columns: Article and Category. The Category specifies one of 14 possible topics, such as politics, sports, or entertainment. It is suitable for classifying detailed and context-rich content. **IIT Patna Reviews**[2] This dataset combines Hindi movie and product reviews with two columns: Review and Sentiment. The Sentiment column has three classes: Positive, Negative, and Neutral. Though primarily used for sentiment analysis, it can also be used for document classification as it deals with user-generated feedback.

For **Extractive Question Answering** task, I have considered datasets IndicQA, MergedQuAD and ChAII. **IndicQA**[7] dataset focuses on question answering in multiple Indian languages, including Hindi. It has four columns: Context, Question, Answer, and AnswerStart. For this task, the Hindi portion of the dataset was used. **MergedQuAD**[9] combines several question-answering datasets, with 90% of its data derived from machine-translated English datasets to Hindi. It has the same structure as IndicQA and includes diverse sources. The machine-translated content introduces unique challenges, such as translation errors or inconsistencies. **ChAII**[10] dataset is from the Kaggle competition "ChAII - Hindi and Tamil Question Answering." For this task, only the Hindi portion (ChAII-Hindi) was used. It shares the structure of IndicQA and focuses on practical, real-world scenarios, making it a valuable resource for benchmarking models in low-resource settings.

Classification			**Question Answering**	
Name	**# Samples**	**# Labels**	**Name**	**# Samples**
BBC Headline	4087	6	MergedQuAD	6,115
BBC News	4335	14	IndicQA	1,052
IIT Patna Reviews	7496	3	ChAII-Hindi	756

Table 1: Dataset statistics for Classification and Question Answering tasks.

3.2 Models

For this study, I have selected two types of language models: Generalized Multilingual Models and specialized Indic language models.

3.2.1 Generalized Multilingual Models

Generalized Multilinugal Models are trained on data from multiple languages across the world, making them versatile for many NLP tasks but not specifically tuned to the intricacies of Indian languages. In this I have considered pre-trained models XLM-RoBERTa and mBERT.

XLM-RoBERTa is a transformer-based language model that trained on 100 languages, including Hindi. It is pre-trained on a massive dataset of multilingual text and is designed to understand a wide variety of languages, including Hindi, and is known for its ability to handle different scripts and language structures. However, it may not capture the finer details of Indian languages such as grammar, dialects, and script-specific nuances.

mBERT is a widely-used multilingual model that trained on 104 languages. It is based on BERT, but unlike the original BERT, which is trained only on English, mBERT can handle multiple languages. While it performs well for general multilingual tasks, its performance on languages like Hindi may not be as specialized since it treats all supported languages equally.

3.2.2 Specialized Indic Models

Specialized Indian Models are specifically trained on Indian languages, which makes them more capable of understanding the unique features of Hindi and other Indian languages, including their scripts, grammar, and dialects. In this, I have taken pre-trained models MuRIL and IndicBERT.

MURIL (Multilingual Representations for Indian Languages) is a model developed by Google specifically for Indian languages. It is pre-trained on text from 17 Indian languages, including Hindi, and is optimized for tasks involving Devanagari and other Indian scripts. MURIL considers the unique structures of Indian languages, making it well-suited for Hindi NLP tasks.

IndicBERT is a lightweight language model pre-trained specifically on Indian languages using a smaller dataset. It is designed to efficiently perform tasks in Indian languages like Hindi. Despite being lightweight, IndicBERT is tailored to handle the grammatical structures, vocabulary, and cultural nuances of Indian languages, making it a strong candidate for comparison.

3.3 Evaluation Settings

For the task of **Document Classification**, all four models are fine-tuned separately on each of the datasets. The performance is evaluated in terms of **accuracy** on the test set. Additionally, I also report the number of **epochs** required during fine-tuning to achieve the best accuracy.

For the task of **Extractive Question Answering**, all four models are fine-tuned on **50% of the MergedQuAD** dataset. The performance is measured using two standard metrics: **Exact Match (EM) Accuracy** and **F1 Score**. Additionally, I also report the number of **epochs** needed to reach these results on the test set.

To ensure a fair comparison, the same fine-tuning framework and hyperparameters (learning rate, batch size, etc.) are used for all models to maintain consistency.

4 Results and Analysis

4.1 Document Classification

[Table 2] Across all datasets for document classification, specialized Indic models (IndicBERT-v2, MuRIL) outperform generalized multilingual models (mBERT, XLM-RoBERTa) indicating their superior capability in handling Hindi text. Generalized models like mBERT and XLM-RoBERTa, trained on diverse languages globally, lack the fine-tuned understanding of Hindi grammar, dialects, and Devanagari script while Specialized Indic models benefit from their training on Indic languages, which allows them to capture subtleties specific to Hindi, such as morphological variations and culturally contextual terms.

IndicBERT-v2 achieves the highest accuracies in all datasets, showcasing the strength of specialization for tasks heavily depend on script and language-specific features. While generalized models, particularly XLM-

RoBERTa, perform reasonably well, but they cannot match the accuracy of Indic models in Hindi document classification tasks. For example, on the BBC Headline dataset, IndicBERT-v2 achieves 77.63%, while XLM-RoBERTa trails behind at 71.6%. This gap exemplifies the advantage of Indic-specific training.

Model	BBC Headline		BBC Article		IIT Patna Reviews	
	# Epochs	Accuracy	# Epochs	Accuracy	# Epochs	Accuracy
mBERT	2	66.04	4	73.13	6	69.98
XLM-RoBERTa	7	71.6	3	76.23	3	74.31
IndicBERT-v2	8	**77.63**	7	**80.16**	3	74.43
MuRIL	4	76.27	8	74.14	8	**75.15**

Table 2: Summary of the results from experiment on Document Classification task, where we evaluate the model's performance in terms of accuracy. The highlighted metrics show the highest performance across the models that are compared for every dataset.

4.2 Extractive Question Answering

[Table 3] Question-answering tasks require models to understand the context, meaning, and relationships between words in the text. Generalized models, like XLM-RoBERTa and mBERT, are trained on a large multilingual corpus, which gives them better reasoning and versatility.

For the MergedQuAD dataset, XLM-RoBERTa performs the best among all models. This is because MergedQuAD contains machine-translated data, and generalized models are better suited for handling such translated datasets.

On the other hand, for IndicQA and ChAII-Hindi, the specialized Indic model MuRIL outperforms all other models. This is due to its training on Indian language datasets, which makes it better at understanding the nuances of Hindi. However, XLM-RoBERTa still gives comparable performance on these datasets, likely because of its larger pretraining corpus.

Surprisingly, IndicBERT-v2 performs poorly in question-answering tasks, with lower EM and F1 scores across all datasets. This could mean that IndicBERT-v2 is not well-optimized for complex tasks like question answering, but it excels in simpler tasks like document classification. Overall, MuRIL shows strong performance in handling context-rich Hindi questions, thanks to its robust pretraining on Indian languages.

Model	# Epochs	MergedQuAD		IndicQA		ChAII-Hindi	
		EM	F1 Score	EM	F1 Score	EM	F1 Score
mBERT	5	64.51	74.11	35.08	47.76	35.25	45.85
XLM-RoBERTa	7	**79.10**	**85.09**	43.82	57.49	37.00	47.41
IndicBERT-v2	4	39.14	54.16	25.95	39.06	20.64	31.88
MuRIL	4	67.99	77.66	**49.43**	**62.83**	**39.01**	**50.47**

Table 3: Summary of the results from experiment on Question Answering task, where we evaluate the model's performance in terms of Exact Match Accuracy (EM) and F1 Score. The highlighted metrics show the highest performance across the models that are compared for every dataset.

5 Conclusion

My experiments clearly show that specialized Indic models like IndicBERT-v2 and MuRIL are better suited for tasks like Document Classification, where understanding Hindi grammar, dialects, and scripts plays a crucial role. Among the models, IndicBERT-v2 stood out with the highest accuracy across all datasets, proving how effective it is for classification tasks. For Question Answering, XLM-RoBERTa performed best on machine-translated datasets, while MuRIL excelled in handling context-rich Hindi questions due to its pretraining on Indian language data. This study highlights the importance of specialized models for Indic languages and the versatility of generalized models for multilingual tasks, offering guidance for model selection in Hindi NLP.

References

[1] Most used languages online by share of websites 2024 | Statista — statista.com. https://www.statista.com/statistics/262946/most-common-languages-on-the-internet/. [Accessed 30-11-2024].

[2] Md Shad Akhtar, Ayush Kumar, Asif Ekbal, and Pushpak Bhattacharyya. A hybrid deep learning architecture for sentiment analysis. In Yuji Matsumoto and Rashmi Prasad, editors, *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 482–493, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[3] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

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

[5] Raj Dabre, Himani Shrotriya, Anoop Kunchukuttan, Ratish Puduppully, Mitesh M Khapra, and Pratyush Kumar. Indicbart: A pre-trained model for indic natural language generation. *arXiv preprint arXiv:2109.02903*, 2021.

[6] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[7] Sumanth Doddapaneni, Rahul Aralikatte, Gowtham Ramesh, Shreya Goyal, Mitesh M Khapra, Anoop Kunchukuttan, and Pratyush Kumar. Towards leaving no indic language behind: Building monolingual corpora, benchmark and models for indic languages. *arXiv preprint arXiv:2212.05409*, 2022.

[8] MIDAS Research Group. Bbc hindi news article classification dataset (nli), 2022. Accessed on November 30, 2024.

[9] Kushal Jain, Adwait Deshpande, Kumar Shridhar, Felix Laumann, and Ayushman Dash. Indic-transformers: An analysis of transformer language models for indian languages. *arXiv preprint arXiv:2011.02323*, 2020.

[10] Kaggle. Chaii - hindi and tamil question answering dataset, 2021. Accessed on November 30, 2024.

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

[12] Nirant Kasliwal. Bbc hindi news headline classification dataset, 2020. Version 0.1, accessed on November 30, 2024.

[13] 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. Muril: Multilingual representations for indian languages, 2021.

[14] BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers,

Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. Bloom: A 176b-parameter open-access multilingual

language model, 2023.

[15] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mt5: A massively multilingual pre-trained text-to-text transformer, 2021.