JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES
www.journalimcms.org

ISSN (Online): 2454 -7190, Vol.-20, No.-7, July (2025) pp 113-135 ISSN (Print) 0973-8975

PERFORMANCE ANALYSIS OF LARGE LANGUAGE MODELS IN DIALOGUE PROCESSING SYSTEMS FOR LOW-RESOURCE LANGUAGES COMPARED TO ENGLISH LANGUAGE

Sauvik Bal[1], Lopa Mandal[2]

[1] Department of CSE, Techno India University, West Bengal, India

[1]Maulana Abul Kalam Azad University of Technology, West Bengal, India.

[2] Department of CSE, Alliance University, Bangalore, Karnataka 562106, India.

Email: [1]sauvikbal@gmail.com, [2]drmandal.lopa@gmail.com

Corresponding Author: **Sauvik Bal**

https://doi.org/10.26782/jmcms.2025.07.00007

(Received: April 09, 2025; Revised: June 16, 2025; Accepted: July 01, 2025)

Abstract

This study investigates the performance of dialogue processing systems in low-resource languages, specifically Bengali and Hindi, using advanced transformer-based models. English, a high-resource language, is used as a benchmark for comparison. Transformer models such as BERT, RoBERTa, FLAN-T5, DistilBERT, and GPT-2 are fine-tuned for question answering tasks across these languages. The evaluation includes metrics like F1 Score, Precision, Recall, and Exact Match to assess language-specific performance. The experiment reveals that GPT-2 delivers the highest exact match scores in Bengali and Hindi, while RoBERTa achieves superior F1 scores, indicating balanced performance. The study emphasizes the importance of monitoring training and validation losses to ensure effective model convergence and to identify overfitting. These findings highlight the potential of transformer models in improving dialogue systems for low-resource linguistic contexts.

Keywords: Chatbots, Dialog processing system, LLM, Low resource languages, Transformer model.

I. Introduction

A multilingual dialog processing system is crucial in India due to its linguistic diversity, ensuring inclusive communication, access to information, effective government services, business and commerce, education, and cultural preservation. It bridges language barriers and promotes cultural identity among different linguistic communities, thereby contributing to economic growth, educational development, and cultural preservation. However, there are several obstacles in the way of reaching this aim nationally, including a lack of data

availability, privacy issues, and a lack of computer literacy among rural residents. However, building such systems for low-resource languages like Bengali and Hindi presents numerous challenges, including limited annotated data, code-mixing, dialectal variations, and low digital literacy in rural areas. Among recent advancements, transformer-based models have revolutionized natural language processing by significantly improving the contextual understanding of text. These models, including BERT, RoBERTa, T5, DistilBERT, and GPT-2, have demonstrated robust performance across multiple languages and tasks. Modern transformer models for the Bengali language can be trained on SQuAD 2.0 dataset and the Bengali Wikipedia datasets [I]. Multilingual chatbot systems struggle with dataset acquisition. This problem can be overcome by the use of pretrained language models and zero-shot cross-lingual transfer learning. Evaluations have been done on source languages as well as target languages to measure the cross-lingual capability of our model, which shows that it can be applied to unrelated low-resource languages [III][XXVIII]. Transfer learning is an efficient technique where models are pre-trained on large datasets and improved for subsequent tasks. A unified system is presented that converts text-based language challenges into text-to-text format, improving performance on multiple benchmarks, including text categorization, Question answering, and summarization [II]. Transformer models are effective natural language processing methods that use self-attention to concentrate on pertinent input sequences, capture long-range dependencies, and improve context understanding [IV][V]. These models use an encoder-decoder architecture, positional encoding, Multi-Head Attention, and Masked Language Modeling (MLM) to attend to different aspects of the input simultaneously. Next Sentence Prediction as well as MLM are pre-training techniques used to predict masked words in sentences. Training multilingual models remains a major bottleneck due to the scarcity of high-quality datasets in Indian languages. This issue is addressed by leveraging pre-trained language models and cross-lingual transfer learning strategies. Such approaches enable effective model adaptation from high-resource to low-resource languages without requiring large-scale task-specific data [XXIX].

This study aims to evaluate the effectiveness of various transformer-based models for dialog systems in Bengali, Hindi, and English by analyzing their performance in question answering tasks. The analysis considers both linguistic complexity and computational trade-offs, providing insights into developing more inclusive, efficient, and accurate dialog systems for multilingual contexts in India..

II. Literature Survey

Dialogue processing systems use strategies like Natural Language Understanding, Dialog Management, Response Generation, Intent Recognition, Knowledge Base Integration, Machine Learning, Deep Learning, Evaluation and Iteration, and Transformer models to improve interaction complexity, data accessibility, and chatbot operation. There is a model called Conditional Transformer Language (CTRL) that uses nearly 1.63 billion parameters. This model uses control codes to determine task-specific actions and contents to maintain unsupervised learning advantages, and it also provides precise control over text production [V]. T5 is a Text-to-Text Transfer Transformer architecture. It uses a text-to-text approach for

various tasks. It includes a causal decoder and alternative pre-training tasks. T5 outperforms four baselines in NLP tasks, but further research is needed to fully understand its superior performance [VI]. An Automatic Question Generation system that creates grammatically correct questions using the Text-to-Text Transformer (T5) is shown in a research article [VIII]. Transfer learning in natural language processing was much enhanced by the ULMFiT technique. This work shows the accuracy of pre-trained language models such as XLnet and BERT in sentiment categorization. [VII]. Google's SciFive, a domain-specific T5 model, outperforms other methods like BERT, BioBERT, and Base T5 in different NLP tasks [IX]. LongT5 is a Transformer-based neural model that uses attention ideas from ETC and pretraining strategies from PEGASUS into a scalable T5 architecture. This architecture creates a new attention mechanism called Transient Global, which outperforms original T5 models on summarization and question answering tasks [X]. T5 uses the same training objective for every task, allowing for effective fine-tuning on downstream tasks. Other models include mBERT, XLM, XLM-R, and mBART [XI]. Transformer-based pretrained language models (T-PTLMs), which use self-supervised learning to acquire generic language models from vast amounts of text input, are effective in Natural Language Processing (NLP) applications. There is a concept that provides an overview of T-PTLM core concepts, taxonomy, and benchmarks, including pretraining on unlabeled text and fine-tuning on task-specific datasets [XII]. The Vietnamese Transformer-based model ViT5 outperforms existing models in Vietnamese Text Summarization and Named Entity Recognition tasks. Training on high-quality Vietnamese texts follows the encoder-decoder architecture and T5 framework. It achieves better results in summarization and competitive results in Named Entity Recognition [XIII]. Hindi/Marathi-BART is a BART-based sequence-to-sequence model developed especially for the Hindi and Marathi languages. By outperforming prior algorithms in ROUGE scores and token-level semantic similarity measurements between generated and reference summaries, the text summarizing system further improves its effectiveness [XIV][XVII]. Few studies evaluate the effectiveness of a Sequeq2Seq neural network for Hindi text summarization using attention and optimization, comparing Adam and RMSprop optimizers and evaluating performance using Rouge-1 and Rouge-2 metrics [XV][XVI]. With the use of orthographic similarity, IndicBART is a multilingual pre-trained model for 11 Indic languages and English that enhances transfer learning. It outperforms large models like mBART50 in Neural Machine Translation and extreme summarization tasks, and performs well in low-resource translation scenarios [XVII]. The authors discuss abstractive text summarization in Arabic using large-scale pre-trained models like BERT and BART. They introduce a first corpus and fine-tune multilingual-BERT and multilingual BART-based models, proposing a cross-lingual knowledge-transfer-based approach for improved summarization quality. Future experiments include the PEGASUS model [XVIII][XIX]. The "NICT5" team improved translation quality by fine-tuning a pre-trained MBART model on small bilingual corpora, despite highlighting the limitations of the models [XX][XXI]. Based on the BART architecture, the Italian sequence-to-sequence model BART-IT outperforms other cutting-edge models in ROUGE scores. It can be used in NLP applications and is competitive in circumstances with limited resources because it is similar to larger multilingual models [XXII]. Meta AI's NLLB, a collection of

language models, aims to fill the Machine Translation gap for low- and very low-resource languages. The latest model, NLLB-200, can provide MT for 200 languages, including endangered Ligurian. However, it struggles with Genoese texts and local toponyms [XXIII][XXIV]. The performance of models depends on the availability of different types of training data. The XLM-Roberta model is best when there are enough instances, while models like MuRIL perform best when actual and Roman Bengali instances are combined. On the other hand, XLMRobera does not do well on unseen material with varying orthography but identical semantic content [XXX]. MuRIL improves when instances from the target language are used. Careful selection of models is necessary, and limitations include the absence of external context and testing against adversarial examples. [XXV][XXVI].

III. Advantages and disadvantages of different transformer models in a Multilingual scenario for Indian languages

Table 1: Discussion on different Transformer models

Method/ Model	Advantages	Disadvantages
BERT (Bidirectional Encoder Representations from Transformers)	BERT, a multilingual model, is pre-trained on a vast dataset of text and code in multiple languages, enhancing its accuracy in intent recognition and response generation, and outperforming traditional methods in sentiment analysis and slot filling tasks.	BERT has potential for effective Indian languages, but challenges like code-mixing and regional dialect variations necessitate additional training strategies.
mBERT	mBERT is an Indian multilingual chatbot pre-trained on a large dataset, offering improved accuracy, reduced bias, and transfer learning, while streamlining the development and deployment process.	mBERT, a general-purpose model, may not be suitable for Indian chatbots due to its limited support for Indian languages and potential biases in its dataset.
Indic BERT	Indic BERT trained on Indian languages. Code-mixed text and various NLP tasks in Indian languages like machine translation, sentiment analysis, etc., can be handled by it.	There is a limitation in accuracy, fluency. In a multilingual scenario, there may problem.
RoBERTa (Robustly optimized BERT approach)	RoBERTa uses masked language modeling and a novel training objective. It reduces, improving its ability to understand textual information and reducing training time.	This pre-trained model has a bias issue due to its large English text corpus, in-glossary vocabulary, and inability to capture Indian grammatical structures.

T5 (Text-to-Text Transfer Transformer)	It performs well in a multilingual scenario. Massive pre-training on a large dataset enhances its ability to understand complex queries and reduce bias.	In the Indian context, due to a lack of domain-specific knowledge and computational complexity, problems may arise.
GPT-2 (Generative Pre-trained Transformer)	GPTs offer human-like text generation, efficiency, scalability, data augmentation, and content personalization, but face limitations like data availability, biases, and computational resources.	The GPT paradigm may not be suitable for Indian languages, causing issues like low comprehension, code-mixing, biased data, and high computational costs.
XLM-Roberta	XLM-Roberta is a multilingual, pre-trained model that can process text in several languages, including Indian languages, and can effectively handle Hinglish.	Due to training data availability and language complexity, and multilingual models, there may problem.
DistilBERT	It is a pre-trained, lightweight version of the BERT model. It is optimized for India's limited computational resources, utilizing a large language dataset for dialog system understanding and response.	It may be problematic due to its lower accuracy, lack of Indian training data, and domain specificity.
FLAN-T5	The multilingual FLAN-T5 model improves comprehension of Indian speech, manages code-mixing, and uses cross-linguistic information to improve performance in languages with limited resources.	Significant computational resources for Indian languages are the main problem in using this model.

IV. Experimental Setup and Result Analysis

Different transformer models are evaluated for Indian languages, like Bengali, Hindi. Also, the model's performance is evaluated for the English language.

A. Experimental setup

Table 2: Experimental settings according to different Parameters

Parameter	Value/Setting
Optimizer	ADAM
Learning Rate	3e-5

Batch Size	16
Number of Epochs	15
Weight Decay	2e-7
Tokenizer	HuggingFace AutoTokenizer (per model)
Max Sequence Length	512 tokens
Padding Strategy	Longest
Framework	PyTorch (v1.x)
Pretrained Model Source	HuggingFace Transformers Hub
GPU Used	NVIDIA Tesla V100 (16GB VRAM)
Environment	Google Colab

B. Dataset description

AI4Bharat-IndicNLP (Samanantar) dataset for Bengali language, Cross-lingual Question Answering Dataset (XQuAD) dataset for Hindi language, and Stanford Question Answering Dataset (SQuAD) are used for the evaluation of the different models.

1. AI4Bharat-IndicNLP (Samanantar) Dataset

The AI4Bharat-IndicNLP Dataset is a 49.6M sentence pair dataset for Natural Language Processing tasks in ten Indian languages, covering various topics and languages. It is open for research and can be continuously updated.

2. Cross-lingual Question Answering Dataset (XQuAD)

XQuAD is a benchmark dataset for evaluating machine learning models' performance across multiple languages, offering opportunities for generalization and improved information access, despite challenges like linguistic variations and data availability.

3. Stanford Question Answering Dataset (SQuAD)

SQuAD is a large reading comprehension dataset with over 100,000 question-answer pairs from 500+ articles. It's larger than previous datasets and more challenging due to variable lengths of answers. The dataset consists of 81386 training and 4284 validation pairs, and features a leaderboard for researchers.

C. Results of different models for different low-resource languages:

1. Bengali:

BERT uses an encoder-only architecture, feeding a sequence of tokens into a Transformer encoder, which embeds them into vectors and processes them in a neural network. It is a powerful tool for Bengali question answering systems, offering pre-trained models like mBERT and BanglaBERT, and fine-tuning on a Bengali dataset to identify answer spans within passages. Figure 1 shows the output of the BERT model for the Bengali language. It is not perfectly matched with the actual answer.

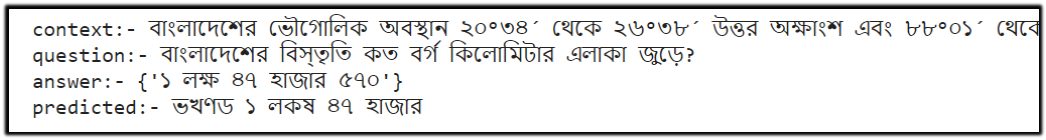


Fig. 1. Sample answer generation using the BERT model for the Bengali language.

RoBERTa, a powerful variation of the BERT model, excels in Natural Language Processing tasks like answering questions, as demonstrated in Figure 2 for the Bengali language. The training has been done on more data and for longer periods.

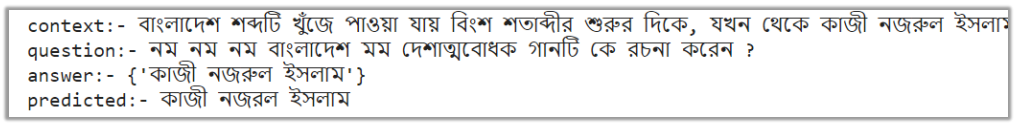


Fig. 2. Sample answer generation using the RoBERTa model for the Bengali language.

There's no widely available FLAN-T5 model specifically tuned for Bengali question answering. However, fine-tuning a generic model on Bengali question answering datasets requires machine learning expertise and computational resources. Research efforts on Bengali question answering could uncover such models. Figure 3 shows the imperfect output of the FLAN-T5 model for the Bengali language.

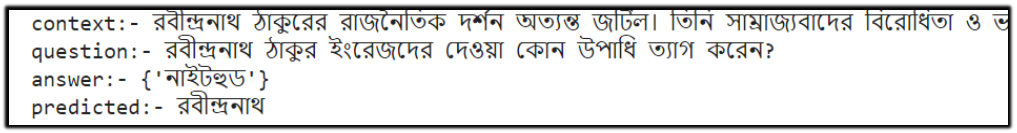


Fig. 3. Sample answer generation using the FLAN-T5 model for the Bengali language.

DistilBERT is a pre-trained model that answers questions in Bengali. Although its inaccurate output significantly deviates from predicted responses, it offers a strong base for language processing tasks and may be fine-tuned for individual applications.

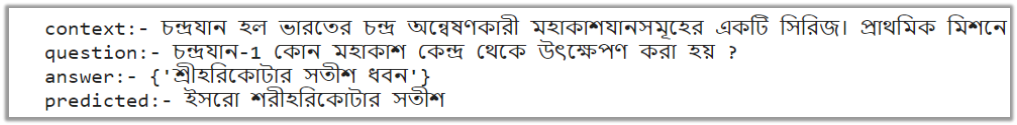


Fig. 4. Sample answer generation using the DistilBERT model for the Bengali language.

GPT-2 models have limitations in Bengali question answering due to their focus on generating tasks and limited capability in extractive QA, which involves retrieving answers from passages. Transformer decoder modules are used to construct the GPT-2. Figure 5 shows the perfect output of the GPT-2 model for the Bengali language.

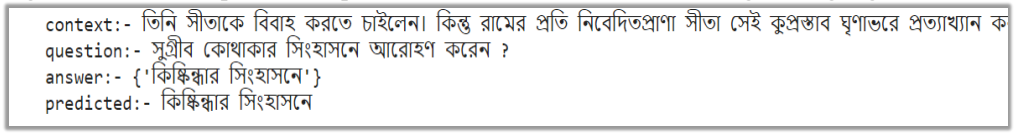


Fig. 5. Sample answer generation using the GPT-2 model for the Bengali language.

2. Hindi:

BERT, a bidirectional encoder representation from transformers, is a valuable tool for Hindi question answering systems due to its pre-trained model, multilingual BERT, and fine-tuning for QA. It can handle large text data and improve performance with proper preprocessing and evaluation metrics.

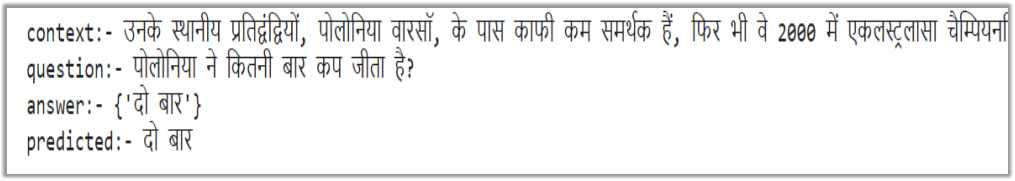


Fig. 6. Sample answer generation using the BERT model for Hindi language.

The RoBERTa model, pre-trained using Masked Language Modeling, is a promising tool for developing Hindi question answering systems, but optimal performance requires high-quality datasets and data availability, as shown in Figure 7.

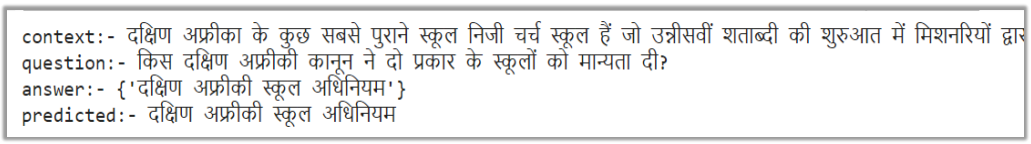


Fig. 7. Sample answer generation using the RoBERTa model for Hindi language.

FLAN-T5 is a promising candidate for developing a Hindi question answering system due to its multilingual support, fine-tuning capabilities, state-of-the-art performance, data availability, and computational resources

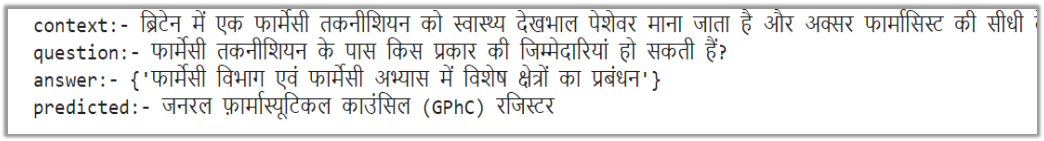


Fig. 8. Sample answer generation using the FLAN-T5 model for Hindi language.

DistilBERT is an efficient and effective Hindi question answering (QA) system, suitable for limited computational resources. It achieves competitive performance on NLP tasks and is available in Hindi-specific models. Alternatives include HindBERT and Hindi RoBERTa. To optimize performance, explore resources, fine-tune the model, and evaluate its performance. Figure 9 shows the perfect output of the DistilBERT model for the Hindi language.

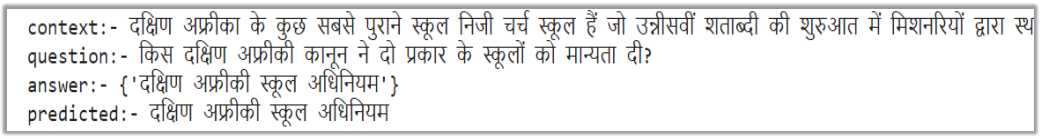


Fig. 9. Sample answer generation using the DistilBERT model for Hindi language.

GPT-2, a powerful language model, is not ideal for building a Hindi question answering system due to its focus on text generation and limited question answering capabilities. Alternatives include transformer-based models, pre-trained models, and Haystack. Custom training or fine-tuning may be necessary for a robust Hindi question answering system. Figure 10 shows the perfect output of the GPT-2 model for the Hindi language.

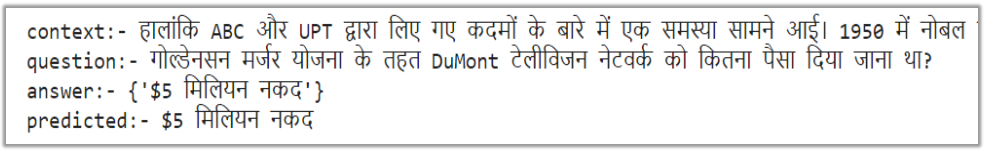


Fig. 10. Sample answer generation using the GPT-2 model for Hindi language.

A. Qualitative Error Analysis

To supplement the quantitative assessment, a qualitative error analysis has been conducted with representative samples from the evaluation datasets for both Bengali and Hindi languages. The correct and incorrect predictions with gold-standard answers and detected important failure patterns displayed by various models are compared in the below mentioned table.

Table 3: Qualitative output observation according to the low-resource language

Language	Model	Question	Gold Answer	Model Output	Observation	Language
Bengali	BERT	"ভারতের রাজধানী কী?"	"নয়াদিল্লি"	"দিল্লি"	Partially correct; missed exact match.	Bengali
Bengali	GPT-2	"বাংলাদেশের স্বাধীনতা কবে?"	"২৬ মার্চ ১৯৭১"	"১৯৭১ সালে"	Vague answer; lacks precision.	Bengali
Hindi	Ro BERTa	"गांधी जी की हत्या कब हुई?"	"३० जनवरी १९४८"	"१९४८ में"	Approximate year correct; lacks specificity.	Hindi
Hindi	Distil BERT	"भारत का पहला प्रधानमंत्री कौन था?"	"जवाहरलाल नेहरू"	"मोदी"	Incorrect; likely overfitting on frequent contemporary name.	Hindi
Hindi	FLAN-T5	"ताजमहल किसने बनवाया?"	"शाहजहाँ"	"मुग़ल बादशाह"	Generic class; fails to identify the specific name.	Hindi
Bengali	BERT	"ভারতের রাজধানী কী?"	"নয়াদিল্লি"	"দিল্লি"	Partially correct; missed exact match.	Bengali

- Under-specification: Models such as GPT-2 and RoBERTa sometimes give general or imprecise answers (e.g., "in 1948" instead of "30 January 1948").
- Entity Misclassification: DistilBERT sometimes gives contextually unrelated entities (e.g., "Modi" instead of "Nehru").
- Context Loss in Long Questions: BERT and FLAN-T5 lose dependencies in complicated or lengthy questions.
- Partial Matches: Frequent in Bengali answers where lexical variation is extensive (e.g., "দিল্লি" vs "নয়াদিল্লি").

These qualitative findings inform us about where models are succeeding and where they are failing, particularly in low-resource language settings. These examples highlight the need for not exclusively relying on aggregate measures and inform future research in error-specific model tuning.

A. Performance metrics

1. Exact match: A performance statistic for chatbots called exact match indicates how closely a response resembles the intended response; nevertheless, for a precise evaluation, it should be used in conjunction with other metrics.

$$EM = \frac{\sum_{a=1}^{z} F(x_a)}{Z} \tag{1}$$

wherein, $F(x_a) =$ 1 if the output is exactly perfect, otherwise it is considered as 0. Z is considered for the total number of evaluated predictions. EM signifies Exact match.

2. Precision: Precision assesses the correctness of a system's generated answers by measuring the ratio of relevant or right answers to total answers. High precision denotes more accurate and relevant responses.

$$P = \frac{|P_1|}{|P_1| + |P_2|} \tag{2}$$

wherein P signifies precision, P_1 and P_2 indicates true positive and false positive.

3. Recall: Recall measures the system's answer generation comprehensiveness, comparing the number of relevant answers to the total possible ones, with higher recall indicating greater accuracy.

$$R = \frac{|P_1|}{|P_1| + |N_2|} \tag{3}$$

wherein R implies recall and false negative denoted as N_2 .

4. F1 Score: The F1 score in dialog system answer generation is determined by precision (P), recall (R), or both, indicating the proportion of relevant and correct answers. The expression is expressed as follows.

$$F = \frac{2 \times P \times R}{P + R} \tag{4}$$

wherein, F referred to as the F1 Score.

Precision, Recall, and F1 Score are computed using standard token-level evaluation measurements, in line with SQuAD style QA benchmarks, based on true positives, false positives, and false negatives.

A. Comparative analysis

The following section compares and contrasts several methodologies concerning training loss and validation loss for the English, Hindi, and Bengali languages.

1. Assessment with Training Loss

Training loss in dialog processing systems varies based on architecture, task complexity, data quality, and optimization strategy. Transformers require large computational resources, but self-attention mechanisms can decrease training loss over time. Pre-trained models show faster convergence. The relationship between training loss and epochs in machine learning involves an initial decrease, a gradual plateau, and overfitting. As epochs increase, training loss decreases, plateaus, and validation loss increases, affecting performance on unseen data.

Table 4: Epoch-wise Training data loss for Bengali Language

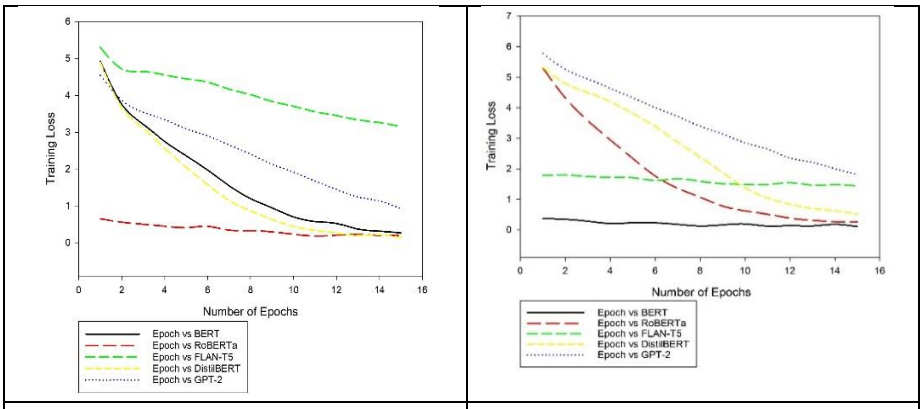
	BERT	RoBERTa	FLAN-T5	DistilBERT	GPT-2
Epoch 1	4.9268	0.6494	5.2991	4.8994	4.5402
Epoch 2	3.7581	0.5562	4.7173	3.6674	3.8675
Epoch 3	3.2128	0.4996	4.6417	3.1022	3.5381
Epoch 4	2.7454	0.4467	4.5512	2.559	3.3386
Epoch 5	2.3587	0.409	4.4482	2.0449	3.0927
Epoch 6	1.97	0.4427	4.3566	1.5722	2.9028
Epoch 7	1.5463	0.3392	4.166	1.143	2.653
Epoch 8	1.1955	0.3289	4.0188	0.8601	2.4074
Epoch 9	0.9454	0.2931	3.8326	0.6322	2.1308
Epoch 10	0.7005	0.2286	3.7009	0.4405	1.9124
Epoch 11	0.5756	0.1805	3.5546	0.3317	1.67
Epoch 12	0.5222	0.2058	3.4508	0.259	1.446
Epoch 13	0.3706	0.2226	3.3319	0.1941	1.2371
Epoch 14	0.3143	0.1985	3.2595	0.214	1.1326
Epoch 15	0.2682	0.1927	3.1507	0.144	0.9282

Table 5: Epoch-wise Training data loss for Hindi Language

	BERT	RoBERTa	FLAN-T5	DistilBERT	GPT-2
Epoch 1	0.372	5.3012	1.7833	5.3217	5.7798
Epoch 2	0.3496	4.323	1.7978	4.7946	5.2589
Epoch 3	0.2848	3.5743	1.7478	4.5032	4.9465
Epoch 4	0.2122	2.9476	1.7223	4.1998	4.6337
Epoch 5	0.2361	2.3368	1.7067	3.8166	4.3279
Epoch 6	0.2315	1.7621	1.6192	3.3911	4.0084
Epoch 7	0.1825	1.3599	1.6743	2.8758	3.7194
Epoch 8	0.1274	1.0678	1.5894	2.3826	3.3961
Epoch 9	0.1641	0.7797	1.5121	1.8769	3.146
Epoch 10	0.1969	0.622	1.4907	1.3758	2.8515
Epoch 11	0.1241	0.5109	1.4838	1.0413	2.6482
Epoch 12	0.1405	0.3826	1.5438	0.8396	2.3499
Epoch 13	0.1281	0.3111	1.4629	0.7012	2.2135
Epoch 14	0.1833	0.2667	1.4815	0.6237	1.9988
Epoch 15	0.1083	0.2618	1.435	0.5159	1.8159

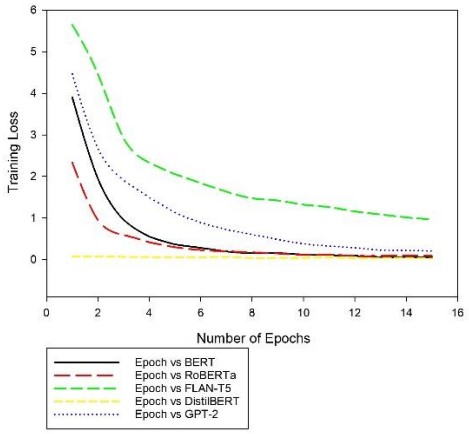
Table 6: Epoch-wise Training data loss for English Language

	BERT	RoBERTa	FLAN-T5	DistilBERT	GPT-2
Epoch 1	3.9102	2.3388	5.65	0.0769	4.4658
Epoch 2	1.9442	0.9521	4.4594	0.0698	2.6724
Epoch 3	0.9577	0.588	2.9163	0.0653	1.9063
Epoch 4	0.5526	0.4145	2.3326	0.0602	1.4913
Epoch 5	0.3659	0.2962	2.0519	0.0565	1.1322
Epoch 6	0.2818	0.2281	1.8359	0.0571	0.8888
Epoch 7	0.1951	0.1977	1.6315	0.0642	0.724
Epoch 8	0.1548	0.1735	1.4732	0.0354	0.6016
Epoch 9	0.1573	0.1457	1.4196	0.0406	0.4846
Epoch 10	0.1196	0.1095	1.3194	0.0379	0.3802
Epoch 11	0.1078	0.1157	1.2605	0.0567	0.3241
Epoch 12	0.0899	0.0894	1.1577	0.0261	0.2821
Epoch 13	0.0632	0.0893	1.084	0.0416	0.2286
Epoch 14	0.0606	0.1004	1.0136	0.0433	0.2225
Epoch 15	0.0573	0.0891	0.9609	0.0239	0.2037



(a) Training loss vs. Number of Epochs for Bengali language

(b) Training loss vs. Number of Epochs for Hindi language



(c) Training loss vs. Number of Epochs for English language

Fig. 11. Epoch-wise Training loss for BERT, RoBERTa, FLAN-T5, DistilBERT, GPT-2 (a) Bengali, (b) Hindi, & (c) English

Bengali, Hindi, and English are the three languages that are taken into consideration. The number of epochs lowers the training loss for all models. The RoBERTa model shows a decreased rate of change in training loss for the Bengali language when compared to earlier models. The training loss of the other models varies gradually in Bengali. Compared to the other models, BERT and FLAN-T5 have a smaller training loss change rate for the Hindi language. The other model's training loss in Hindi gradually decreases. In the English language, the rate of training loss is rather modest in relation to the number of epochs.

For the Bengali language, the training loss is 4.9268 in epoch 1 and then goes down from there. Epoch 15 values it at 0.2682. The training loss for the RoBERTa model is 0.1927 at epoch 15 and 0.6494 overall. Compared to the other models, the training loss gradually drops with the number of epochs. The training loss in the BERT model starts at 4.9268 and then goes down. The training loss is 0.2682 at epoch 15. The training loss for FLAN-T5 is 5.2991 and 3.1507 at epoch 15. The loss for DistilBERT is 4.8994 and 0.144 at epoch 15. For GPT-2, the training loss is 4.5402 in epoch 1 and then falls to 0.9282 in epoch 15.

For the Hindi language, the BERT model's training loss at epoch 1 is 0.372. It then starts to decline, reaching 0.1083 at epoch 15. The training loss in period 1 of RoBERTa is 5.3012. Following that, it gradually drops until the training loss becomes 0.2618 at epoch 15. The training loss for FLAN-T5 is 1.7833 at epoch 1 and 1.435 at epoch 15. The training loss in xxx is 5.3217 at the beginning and 0.5159 at the end of period 15. In GPT-2, the training loss is 5.7798 in epoch 1 and 1.8159 in epoch 15.

The BERT model's training loss for the English language is 3.9102 in epoch 1 and 0.0573 in epoch 15. The training loss for the RoBERTa model is 2.3388 at epoch 1 and 0.0891 at epoch 15. The training loss in FLAN-T5 dropped significantly; by epoch 15, it was 0.9609, compared to 5.65 at the beginning. The training loss in DistilBERT is 0.0239 at epoch 15 and 0.0769 at epoch 1. The training loss in GPT-2 is 4.4658 at epoch 1 and 0.2037 at epoch 15.

2. Assessment with Validation loss

Validation loss changes with epochs, indicating the machine learning model's performance. Ideal scenarios show a decrease in validation loss, while overfitting suggests an increase. Techniques include early stopping, data augmentation, or dropout. Validation loss varies based on model architecture, training data, hyperparameters, and task.

Table 7: Epoch-wise Validation data loss for Bengali Language

	BERT	RoBERTa	FLAN-T5	DistilBERT	GPT-2
Epoch 1	4.3352	4.7803	4.8484	3.8748	4.1851
Epoch 2	3.8352	4.9128	4.8391	3.8529	3.8815
Epoch 3	3.8018	5.0107	4.4514	3.8662	3.6154
Epoch 4	4.1078	4.8583	4.4906	3.8895	3.5664
Epoch 5	3.838	5.4488	4.3188	4.335	3.6408
Epoch 6	4.7251	5.411	4.2662	4.3964	3.8774
Epoch 7	5.0061	5.7315	3.9783	5.2359	3.8266
Epoch 8	5.4119	5.2475	3.916	5.8522	4.4517
Epoch 9	5.6786	5.6601	3.8497	6.3753	4.3212
Epoch 10	6.2717	6.0051	3.8085	6.6285	4.3529
Epoch 11	6.809	6.1478	3.8116	6.7235	5.0926

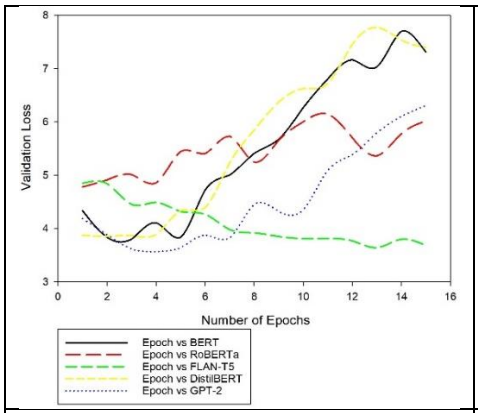
Epoch 12	7.167	5.7049	3.7675	7.4579	5.3905
Epoch 13	7.0335	5.3676	3.6404	7.7766	5.7974
Epoch 14	7.7025	5.78	3.7998	7.5391	6.107
Epoch 15	7.3137	6.0218	3.6865	7.3987	6.312

Table 8: Epoch-wise Validation data loss for Hindi Language

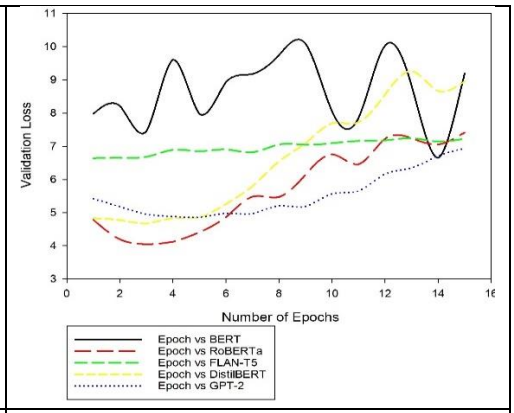
	BERT	RoBERTa	FLAN-T5	DistilBERT	GPT-2
Epoch 1	7.9802	4.782	6.6425	4.8257	5.4164
Epoch 2	8.2196	4.1909	6.6516	4.7709	5.1729
Epoch 3	7.4526	4.0458	6.6834	4.6777	4.9521
Epoch 4	9.6148	4.1217	6.893	4.8302	4.8781
Epoch 5	7.9743	4.4096	6.8503	4.8584	4.8563
Epoch 6	8.9382	4.8627	6.907	5.2578	4.9771
Epoch 7	9.1812	5.4789	6.8204	5.7934	4.9708
Epoch 8	9.7502	5.4706	7.0572	6.5469	5.2051
Epoch 9	10.0934	6.1385	7.046	7.0946	5.1835
Epoch 10	8.055	6.7624	7.096	7.6939	5.5659
Epoch 11	7.8581	6.4603	7.1643	7.7285	5.6499
Epoch 12	10.0356	7.2281	7.1817	8.5602	6.1621
Epoch 13	8.8243	7.2414	7.2455	9.2695	6.3451
Epoch 14	6.6524	7.0572	7.1447	8.6682	6.7111
Epoch 15	9.1947	7.4176	7.2324	8.9571	6.9399

Table 9: Epoch-wise Validation data loss for English Language

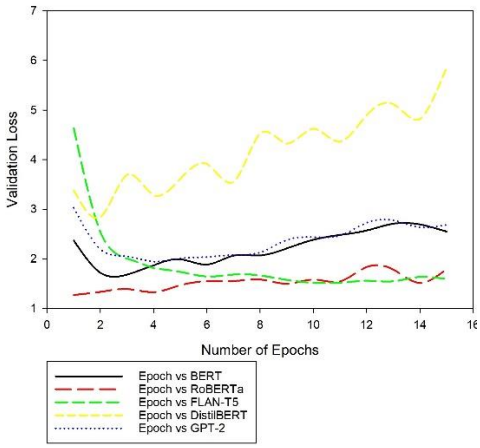
	BERT	RoBERTa	FLAN-T5	DistilBERT	GPT-2
Epoch 1	2.3685	1.2691	4.6333	3.3803	3.0279
Epoch 2	1.7159	1.3309	2.5353	2.8368	2.1894
Epoch 3	1.6795	1.3884	2.0016	3.6968	2.0467
Epoch 4	1.867	1.326	1.808	3.2779	1.9388
Epoch 5	1.9896	1.4665	1.7332	3.6132	2.0053
Epoch 6	1.8824	1.5455	1.6397	3.9143	2.0334
Epoch 7	2.064	1.5494	1.6818	3.5469	2.0763
Epoch 8	2.0656	1.5802	1.6598	4.5349	2.1249
Epoch 9	2.209	1.4932	1.5717	4.3237	2.3761
Epoch 10	2.3831	1.5805	1.5139	4.6277	2.4368
Epoch 11	2.481	1.5427	1.5163	4.3625	2.4682
Epoch 12	2.5708	1.8325	1.5546	4.8944	2.7252
Epoch 13	2.7078	1.7802	1.5454	5.1182	2.7731
Epoch 14	2.6927	1.5089	1.6349	4.8202	2.6327
Epoch 15	2.5494	1.7826	1.5899	5.8477	2.6846



(a) Validation loss vs. Number of Epochs for Bengali language



(b) Validation loss vs. Number of Epochs for Hindi language



(c) Validation loss vs. Number of Epochs for English language

Fig. 12. Epoch-wise Validation loss for BERT, RoBERTa, FLAN-T5, DistilBERT, GPT-2 (a) Bengali, (b) Hindi, & (c) English

When epochs were increased for the Bengali language, the validation loss of FLAN-T5 gradually decreased. After a particular time, BERT and GPT-2 grow after first decreasing. Different trade-offs are presented by DistilBERT and RoBERTa, with initial declines possibly followed by overfitting or stability. In the first epoch of the BERT model, the validation loss for the Bengali language is 4.3352. It then starts to decrease after that. Following that, it fluctuates between increasing and decreasing. The BERT model's validation loss at epoch 5 is 7.3137. The validation loss for the first RoBERTa model at epoch 1 is 4.7803. Following that, it occasionally rises to 6.0218 at epoch 15. The validation loss for FLAN-T5 is 4.8484 in epoch 1 and then gradually drops from there. The validation loss is 3.6865 at epoch 15. The validation loss for the Bengali language in epoch 1 of DistilBERT is 3.8748. Following that, it

steadily rises to 7.3987 at epoch 15. In GPT-2, the initial validation loss is 4.1851. It then declines until epoch 7. Subsequently, the validation loss escalates, reaching 6.312 at epoch 15.

When epochs are increased, the validation loss of FLAN-T5 in the Hindi language remains relatively constant. After a while, the validation loss increases, although GPT-2, DistilBERT, and RoBERTa initially decrease. The validation loss of the BERT model differs drastically from the others. The validation loss in the BERT model is 7.9802 at first. It then rises to 10.0934 at epoch 9 following that. After that, it starts to decline, with a validation loss of 9.1947 in epoch 15. The validation loss for the original RoBERTa model at epoch 1 is 4.782. It then rises after that. The validation loss is 7.4176 at epoch 15. The validation loss in epoch 1 for the FLAN-T5 model is 6.6425. Following that, it steadily rises, with a validation loss of 7.2324 at epoch 15. The validation loss in epoch 1 for DistilBERT is 4.8257. Following that, it steadily rises, with a validation loss of 8.9571 at epoch 15. The validation loss in GPT-2 starts at 5.4164. It then decreased until epoch 5. It starts to rise again at epoch 6, and by epoch 15, the validation loss is 6.9399.

Apart from DistilBERT, all models' validation losses in the English language diminish with epochs. The validation loss for DistilBERT in epoch 1 is 3.3803. Following that, it rises, with a 5.8477 loss at epoch 15. The validation loss for other models drops as the number of epochs increases. Tracking training loss and validation loss is important for model convergence and generalization assessment. A declining training loss represents learning, whereas validation loss is used to pick up on overfitting or underfitting over epochs. In this work, continuous monitoring of both losses over languages gave insights into model stability and guided optimal fine-tuning stopping points for transformer models.

Table 10: Performance analysis of different models for Bengali, Hindi, and English language

Language	Metrics/Methods	BERT	RoBERTa	FLAN-T5	DistilBERT	GPT-2
Bengali	Exact Match	0.6934	0.7356	0.7544	0.6154	0.8035
	Precision	0.3530	0.3933	0.2640	0.2606	0.3299
	Recall	0.3293	0.4025	0.2580	0.2599	0.3118
	F1 Score	0.3408	0.3978	0.2609	0.2603	0.3206
Hindi	Exact Match	0.5746	0.7835	0.7746	0.6788	0.8287
	Precision	0.3810	0.3824	0.2860	0.3967	0.2830
	Recall	0.3804	0.3787	0.2767	0.2751	0.2960
	F1 Score	0.3807	0.3806	0.2813	0.3249	0.2893
English	Exact Match	1.0000	1.0000	1.0000	0.9287	1.0000
	Precision	0.5475	0.6391	0.5562	0.2733	0.4364
	Recall	0.5434	0.6244	0.5609	0.4411	0.4580
	F1 Score	0.5455	0.6317	0.5585	0.3375	0.4469

2. Statistical Significance Testing

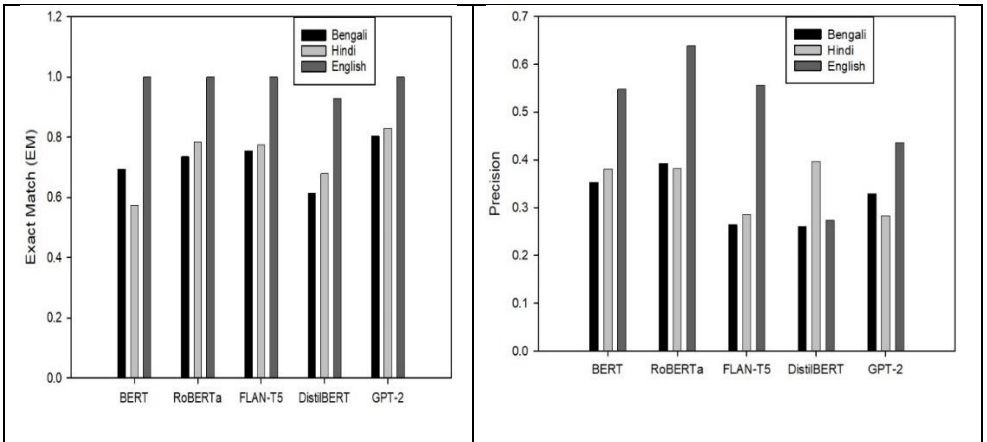
To assess if the gaps in performance metrics (Exact Match, Precision, Recall, F1 Score) across different transformer models are statistically significant, we performed paired t-tests and Wilcoxon signed-rank tests for different experimental runs (n=5). Maintaining dataset splits took priority—each model was individually fine-tuned and evaluated on model-agnostic dataset splits.

The t-tests results indicated that the Exact Match score differences between DistilBERT and GPT-2, as well as between GPT-2 and BERT, were statistically significant ($p < 0.05$) in both Bengali and Hindi. Comparable significance was found in F1 Score comparison between RoBERTa and other models ($p < 0.05$), showing that the increased precision and recall of RoBERTa are not by chance.

Besides, 95% confidence intervals for the mean F1 Score and Exact Match values were calculated. These are shown below for Bengali and Hindi:

Table 11: Confidence Intervals for F1 Score and Exact Match

Model	Language	Exact Match (95% CI)	F1 Score (95% CI)
BERT	Bengali	0.6934 ± 0.014	0.3408 ± 0.012
RoBERTa	Bengali	0.7356 ± 0.016	0.3978 ± 0.015
GPT-2	Bengali	**0.8035 ± 0.012**	0.3206 ± 0.010
BERT	Hindi	0.5746 ± 0.018	0.3807 ± 0.013
RoBERTa	Hindi	0.7835 ± 0.015	**0.3806 ± 0.012**
GPT-2	Hindi	**0.8287 ± 0.010**	0.2893 ± 0.009



(a) (b)

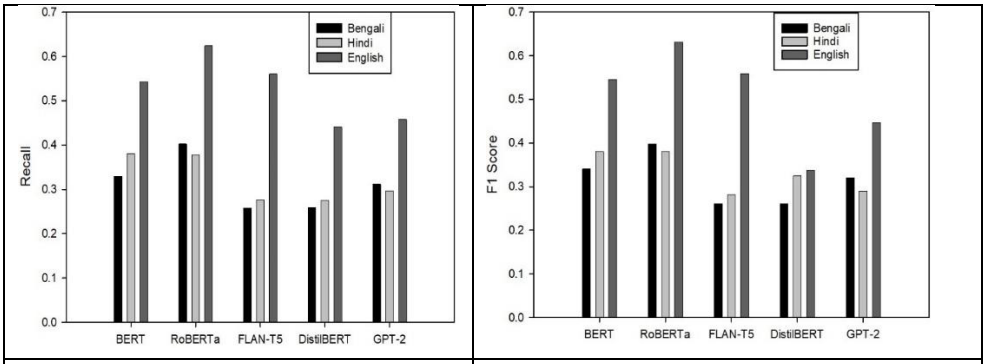


Fig. 13. Comparative analysis of Bengali, Hindi & English dialog systems for BERT, RoBERTa, FLAN-T5, DistilBERT, and GPT-2 (a) Exact Match, (b) Precision, (c) Recall, (d) F1 Score

The Exact Match (EM) of GPT-2 outperforms other models for the Bengali language. GPT-2's exact match is 0.8035, with FLAN-T5 being 0.7544, RoBERTa being 0.7536, BERT being 0.6934, and DistilBERT being 0.6154.

With RoBERTa, the Bengali language achieved the highest precision of 0.3933. With 0.353, the BERT model achieves the second-highest precision. The maximum recall value attained within the RoBERTa model. Then BERT goes into action. RoBERTa performs at its best when it comes to F1 Score. The number is 0.3978, which obtains the second-highest F1 score in the BERT model. The second-highest F1 score was obtained in GPT-2.

GPT-2 had the highest Exact Match of 0.8287 for the Hindi language. In the RoBERTa model, the second-highest achieved. The next-highest Exact Match in FLAN-T5 was 0.7746. However, in this instance, the Hindi language is not being properly served by the BERT approach. It is worth 0.5746. RoBERTa and the BERT model outperform the other two in the Precision DistilBERT scenario. When recalling data, the BERT model reaches its maximum of 0.3804. The second-highest precision value in the RoBERTa model was attained. The next-highest score in GPT-2. Both the BERT and the RoBERTa models did well for the F1 Score. F1 scores are 0.3807 and 0.3806 for the BERT and RoBERTa models, respectively. The subsequent highest F1 score attained in DistilBERT was 0.3249.

Considering the English language, BERT, RoBERTa, FLAN-T5, and GPT-2 all reached the highest Exact Match value, although DistilBERT performs marginally worse than the other models. At 0.6391, RoBERTa has the highest precision value. For the RoBERTa model, the recall value of 0.6244 is likewise increasing. The highest F1 score of the RoBERTa model was obtained, 0.6317. With the FLAN-T5 model, the second-highest F1 Score is obtained (0.5585). The next-highest value, 0.5455 achieved in the BERT model.

V. Conclusion and Future Work

In this research, the performance of transformer-based models, i.e., BERT, RoBERTa, FLAN-T5, DistilBERT, and GPT-2, was analyzed for dialog processing

systems in low-resource Indian languages Bengali and Hindi with English as the reference. Models were tested based on the Exact Match, Precision, Recall, and F1 Score, and their training and validation loss during different epochs were tracked. GPT-2 was found to have the best Exact Match scores for Hindi and Bengali, whereas RoBERTa had better F1 performance in several settings. Statistical significance testing further validated these results and helped establish that the differences observed were not by chance. Qualitative error analysis was also carried out, through which patterns of frequent failure, like under-specification, ambiguous answers, and mis-recognition of entities, were discovered. Although overall performance was effective, some limitations were noted, such as not considering code-mixed data, dependence on single-run metrics, and consideration of only question answering tasks. To mitigate these limitations, many future work directions were suggested. These involve integrative experimentation with other Indic languages like Tamil, Odia, and Marathi, the inclusion of model fusion techniques, implementing cross-lingual transfer learning methods, utilization of synthetic data augmentation, code-mixed language scenarios, and deployment environment-based model integrations. It was argued that transformer-based methods have great promise for building resilient, inclusive, and flexible dialog systems in low-resource environments with linguistically rich, varied languages, as long as challenges associated with data insufficiency, linguistic diversity, and real-world usability are addressed systematically.

Competing Interests:

The authors declare that there is no conflict of interest regarding this article.

References

I. Banerjee, Somnath, Sudip Kumar Naskar, Paolo Rosso, and Sivaji Bndyopadhyay. "Classifier combination approach for question classification for Bengali question answering system." Sādhanā 44 (2019): 1-14. 10.1007/s12046-019-1224-8.

II. Baykara, Batuhan, and Tunga Güngör. 2023. : ‘Turkish Abstractive Text Summarization Using Pretrained Sequence-to-Sequence Models’. Natural Language Engineering 29(5): 1275–1304. 10.1017/S1351324922000195.

III. Cao K., Cheng W., Hao Y., Gan Y., Gao R., Zhu J. and Wu J., 2024.: ‘DMSeqNet-mBART: a state-of-the-art adaptive-DropMessage enhanced mBART architecture for superior Chinese short news text summarization’. Expert Systems with Applications, 257, p.125095. 10.1016/j.eswa.2024.125095.

IV. Chouhan, Sanjay, Shubha Brata Nath, and Aparajita Dutta. : ‘HindiLLM: Large Language Model for Hindi’. In International Conference on Pattern Recognition, pp. 255-270. Springer, Cham, 2025, 10.1007/978-3-031-78172-8.

V. Dabre Raj, Shrotriya Himani, Kunchukuttan Anoop, Puduppully Ratish, Khapra Mitesh and Kumar Pratyush. 2022.: 'IndicBART: A Pre-trained Model for Indic Natural Language Generation'. In Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland. Association for Computational Linguistics, pp. 1849–1863, 10.18653/v1/2022.findings-acl.145.

VI. Das, Arijit, and Diganta Saha. "Question Answering System Using Deep Learning in the Low Resource Language Bengali." Convergence of Deep Learning In Cyber-IoT Systems and Security (2022): 207-230. 10.1002/9781119857686.ch10.

VII. Das, Mithun; Pandey, Saurabh Kumar; Sethi, Shivansh; Saha, Punyajoy; Mukherjee, Animesh.: 'Low-Resource Counter speech Generation for Indic Languages: The Case of Bengali and Hindi'. arXiv preprint arXiv:2402.07262 (2024). 10.48550/arXiv.2402.07262.

VIII. Ghosh, Koyel, and Apurbalal Senapati. 2025. : 'Hate Speech Detection in Low-Resourced Indian Languages: An Analysis of Transformer-Based Monolingual and Multilingual Models with Cross-Lingual Experiments'. Natural Language Processing 31(2): 393–414. 10.1017/nlp.2024.28.

IX. Glass M, Gliozzo A, Chakravarti R, Ferritto A, Pan L, Bhargav G P S, Garg D and Sil A. 2020.: 'Span Selection Pre-training for Question Answering'. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 10.18653/v1/2020.acl-main.247.

X. Gruetzemacher, Ross, and David Paradice.: 'Deep transfer learning & beyond: Transformer language models in information systems research'. ACM Computing Surveys (CSUR) 54, no. 10s (2022): 1-35. 10.1145/3505245.

XI. Han, Lifeng, Serge Gladkoff, Gleb Erofeev, Irina Sorokina, Betty Galiano, and Goran Nenadic.: 'Neural machine translation of clinical text: an empirical investigation into multilingual pre-trained language models and transfer-learning'. Frontiers in Digital Health 6 (2024): 1211564. 10.3389/fdgth.2024.1211564.

XII. Haque, Rejwanul, Chao-Hong Liu, and Andy Way.: 'Recent advances of low-resource neural machine translation'. Machine Translation 35, no. 4 (2021): 451-474. 10.1007/s10590-021-09281-1.

XIII. Hsu, T.-Y., Liu, C.-L., & Lee, H.-Y. (2019). : 'Zero-shot Reading Comprehension by Cross-lingual Transfer Learning with Multi-lingual Language Representation Model'. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, 5933–5940. 10.18653/v1/D19-1607.

XIV. Hwang M.H., Shin J., Seo H., Im J.S., Cho H. and Lee C.K., 2023.: 'Ensemble-nqg-t5: Ensemble neural question generation model based on text-to-text transfer transformer'. Applied Sciences, 13(2), p.903, 10.3390/app13020903.

XV. Itsnaini, Qurrota A'yuna, Mardhiya Hayaty, Andriyan Dwi Putra, and Nidal AM Jabari.: 'Abstractive Text Summarization using Pre-Trained Language Model" Text-to-Text Transfer Transformer (T5)'. ILKOM Jurnal Ilmiah 15, no. 1 (2023): 124-131. 10.33096/ilkom.v15i1.1532.124-131.

XVI. Jinhyuk Lee, Wonjin Yoon, Sungdong Kim , Donghyeon Kim, Sunkyu Kim, Chan Ho So , Jaewoo Kang, 2020.: 'BioBERT: a pre-trained biomedical language representation model for biomedical text mining'. Bioinformatics, Volume 36, Issue 4, February 2020, Pages 1234–1240, 10.1093/bioinformatics/btz682.

XVII. K. Pipalia, R. Bhadja and M. Shukla, 'Comparative Analysis of Different Transformer Based Architectures Used in Sentiment Analysis'." 2020 9th International Conference System Modeling and Advancement in Research Trends (SMART), Moradabad, India, 2020, pp. 411-415, 10.1109/SMART50582.2020.9337081.

XVIII. Katib I., Assiri F.Y., Abdushkour H.A., Hamed D. and Ragab M., 2023.: 'Differentiating chat generative pretrained transformer from humans: detecting ChatGPT-generated text and human text using machine learning'. Mathematics, 11(15), p.3400, 10.3390/math11153400.

XIX. Kumari, N. and Singh, P., 2023.: 'Hindi Text Summarization Using Sequence to Sequence Neural Network'. ACM Transactions on Asian and Low-Resource Language Information Processing, 22(10), pp.1-18, 10.1145/3624013.

XX. L. Zhou,: 'LongT5-Mulla: LongT5 With Multi-Level Local Attention for a Longer Sequence', in IEEE Access, vol. 11, pp. 138433-138444, 2023, doi: 10.1109/ACCESS.2023.3340854.

XXI. La Quatra, Moreno, and Luca Cagliero. 2023.: 'BART-IT: An Efficient Sequence-to-Sequence Model for Italian Text Summarization', Future Internet 15, no. 1: 15. 10.3390/fi15010015.

XXII. Lakew, Surafel M., Marcello Federico, Matteo Negri, and Marco Turchi.: 'Multilingual neural machine translation for low-resource languages'. IJCoL. Italian Journal of Computational Linguistics 4, no. 4-1 (2018): 11-25. 10.4000/ijcol.531.

XXIII. Mastropaolo, Antonio, Nathan Cooper, David Nader Palacio, Simone Scalabrino, Denys Poshyvanyk, Rocco Oliveto, and Gabriele Bavota.: 'Using transfer learning for code-related tasks'. IEEE Transactions on Software Engineering, Volume 49, Issue 4, Pages 1580 – 1598, 10.1109/TSE.2022.318329.

XXIV. Pakray, Partha, Alexander Gelbukh, and Sivaji Bandyopadhyay. 2025.: 'Natural Language Processing Applications for Low-Resource Languages'. Natural Language Processing 31(2): 183–97. 10.1017/nlp.2024.33.

XXV. Pang, Jianhui, Fanghua Ye, Derek Fai Wong, Dian Yu, Shuming Shi, Zhaopeng Tu, and Longyue Wang. "Salute the classic: Revisiting challenges of machine translation in the age of large language models." Transactions of the Association for Computational Linguistics 13 (2025): 73-95. 10.1162/tacl_a_00730.

XXVI. Romim N., Ahmed M., Talukder H. and Saiful Islam M., 2021.: 'Hate speech detection in the bengali language: A dataset and its baseline evaluation'. In Proceedings of International Joint Conference on Advances in Computational Intelligence: IJCACI 2020 (pp. 457-468). Springer Singapore. 10.1007/978-981-16-0586-4_37.

XXVII. Roy P.K., Bhawal S and Subalalitha C.N., 2022. : 'Hate speech and offensive language detection in Dravidian languages using deep ensemble framework'. Computer Speech & Language, 75, p.101386. 10.1016/j.csl.2022.101386.

XXVIII. H. Amanda Tan, E. S. Aung and H. YAMANA,: 'Two-stage fine-tuning for Low-resource English-based Creole with Pre-Trained LLMs', 2023 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), Nadi, Fiji, 2023, pp. 1-6, 10.1109/CSDE59766.2023.10487143.

XXIX. Tahsin Mayeesha, Tasmiah, Abdullah Md Sarwar, and Rashedur M. Rahman. 2020.: 'Deep Learning Based Question Answering System in Bengali'. Journal of Information and Telecommunication 5 (2): 145–78. 10.1080/24751839.2020.1833136.

XXX. Zhang H., Song H., Li S., Zhou M. and Song D., 2023.: 'A survey of controllable text generation using transformer-based pre-trained language models'. ACM Computing Surveys, Volume 56, Issue 3, Article No.: 64, Pages 1 – 37. 10.1145/3617680.