BengaliBot: Neural Conversations Unleashed

Shabab Abdullah, Jannatul Ferdoshi, Asit Kumar Halder, Taskia Siddika and Shuria Akter Ethuna
Department of Computer Science and Engineering (CSE)
School of Data and Sciences (SDS)
Brac University

***Abstract*—BengaliBot is a state-of-the-art chatbot project that uses neural networks to transform the development and understanding of Bengali natural language. Our primary objective is to create a sophisticated conversational agent that can respond contextually, manage scenarios in advance, and provide seamless language support—including support for several dialects. BengaliBot employs cutting-edge neural architectures, including transformers and recurrent neural networks (RNNs), to attain an accuracy rate of at least 90 percent in understanding human input and preserving context throughout discussions. The development of a strong natural language understanding (NLU) model trained on a variety of Bengali datasets is the first of this project's many essential elements. BengaliBot uses neural language generation models to deliver pre-written answers to frequently asked questions, greetings, and scenarios. BengaliBot incorporates an optional learning mechanism that refines replies depending on user interactions, resulting in quantifiable improvements in conversation quality over time. This significantly improves the quality of discussions. The chatbot may learn new things on a constant basis and provide more precise and tailored replies thanks to this adaptive technique. An innovative attempt at natural language processing in Bengali, BengaliBot shows how neural networks may be used to build multilingual, intelligent, and adaptable companions. Through the advancement of AI-powered conversational interfaces in Bengali, this research creates new opportunities for context-aware and interactive applications. BengaliBot is expected to have a major impact on the area and open the door to improved Bengali language human-computer interactions.**

***Index Terms*—Machine Learning, Deep learning, RNN, Natural Language Processing(NLP), Bert**

I. Introduction

The combination of natural language processing has made huge steps forward in the field of artificial intelligence, which is always changing. With the help of cutting-edge neural networks, the BengaliBot project is a ground-breaking effort that could change the way people talk over the phone in Bengali. BengaliBot is a shining example of innovation because it solves the difficult problems that come with reading and writing complex Bengali text. With a major focus on natural language understanding (NLU), keeping the conversation in context, and supporting multiple languages, BengaliBot wants to push the limits of language AI.

BengaliBot is a neural language model that is trying to understand human input with a minimum success rate of 90percent. It does this by using both recurrent neural networks (RNNs) and transformers. One feature that stands out is how well it keeps talks on track and on topic, making sure that replies not only answer questions but also move naturally through the conversation.

BengaliBot is more useful because it has prepared answers for typical talking situations. It uses neural language generation models to mimic dynamic and context-aware conversations, which makes relations between users better. By supporting normal Bengali and at least three important accents, the project promises to be open to different languages and encourages everyone to use the same interface.

The project is very forward-thinking because it includes an alternative learning system that lets the robot change and get better at responding over time based on how people use it.With its wide range of features, BengaliBot not only wants to improve the Bengali user experience, but it also wants to show how neural networks can change the field of talking AI. This introduction sets the stage for a detailed look at the project's structure, methods, and expected results. It also gives a sneak peek at the exciting possibilities that BengaliBot brings to the head of Bengali language processing.

II. Working with Dataset

Our dataset curation process involves a continuous cycle of improvement. We iterate on the dataset, refining it to capture the evolving nuances of language and incorporate the latest trends and expressions. This dynamic approach ensures that our Bangla Chatbot stays up-to-date and relevant in a rapidly changing linguistic landscape.

Beyond translation, we prioritize the cultural context of Bengali language usage. This involves not only linguistic nuances but also cultural references and contextual understanding unique to Bengali speakers. By infusing our dataset with cultural relevance, we enhance the authenticity of our Chatbot's responses and foster a deeper connection with users.

In handling linguistic elements, we pay special attention to idioms, colloquialisms, and informal language commonly used in everyday conversations. This attention to detail allows our Chatbot to not only understand formal queries but also engage in casual and natural interactions, mimicking the way people naturally communicate.

Moreover, our dataset covers a diverse range of topics, ensuring that the Bangla Chatbot is well-equipped to handle a wide spectrum of user queries. This diversity helps the model generalize its knowledge and respond effectively across various domains, enhancing its versatility.

Throughout the data preparation workflow, quality control measures are implemented to identify and rectify inconsistencies. This includes addressing biases that may arise during the translation process or biases present in the original English

datasets, promoting fairness and impartiality in the Chatbot's responses.

In summary, our meticulous dataset construction and refinement process, coupled with a focus on cultural context and linguistic diversity, contribute to the creation of a Bangla Chatbot with strong language skills, cultural sensitivity, and the ability to engage users in a meaningful and authentic manner.

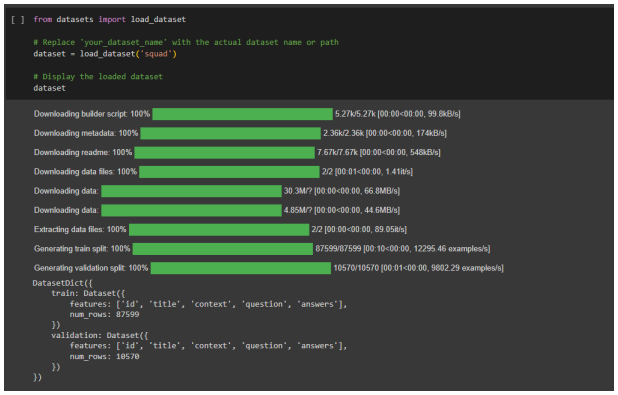

Fig. 1. Dataset

III. Data Preprocessing

When working on the BengaliBot project, the data preparation step is very important because it turns raw information into a file that can be used for training and testing. The Squad version 2.0 model's parameters—id, title, context, question, and answer—go through a number of preparation steps that make neural network training more effective. For Bengali writing, tokenization is the process of splitting it up into single words or sub-word groups. This helps the model understand how language works and makes it better at coming up with clear answers.

Stopword Handling: Common Bengali stopwords are found and taken out to get rid of noise and let the model focus on more important parts of the language. When it's needed, stemming is used to break down words into their root forms. This helps get to the important meaning information with a smaller vocabulary. Cleaning Special Characters: The text is cleaned of special characters, marks, and non-alphanumeric symbols so that the model doesn't learn trends that aren't important from noise.

Normalization: Text normalization methods deal with differences in writing, case, and style. This makes the information more consistent and boosts the model's ability to generalize.

Balancing: Methods like oversampling and undersampling fix class imbalances by making sure that all language variations and situations are fairly represented.

Data Augmentation (when possible): Back-translation or paraphrasing are two techniques that can be used to add to the dataset and give the model a wider range of language patterns and situations to deal with. The Squad version 2.0 model's properties are improved and made better for training neural networks through these preparation steps. After the information was cleaned up and made consistent, it was used to build the models that BengaliBot uses to understand normal language, keep conversations on track, and come up with responses.

IV. Methodology

The methodology utilized in the development of BengaliBot is a methodical and all-encompassing strategy with the objective of furnishing an effective and versatile conversational AI model. During the phase of dataset acquisition, the Bengali-translated Squad version 2.0 dataset is utilized. The dataset comprises critical attributes, including ID, title, context, query, and answer. In order to augment the diversity of the dataset, the Google API for translation is utilized, thereby guaranteeing the inclusion of instances in both English and Bengali language contexts.

Data preprocessing is a crucial step to refine the dataset for training and evaluation. This involves several key processes, including tokenization to break Bengali text into single words or smaller groups, stopword handling to remove common Bengali stopwords, and stemming to reduce words to their root form for capturing essential semantic information. Additionally, special character cleaning is performed to eliminate punctuation and non-alphanumeric symbols, and normalization techniques are applied to handle variations in spelling, case, and format, promoting consistency across the dataset.

The model architecture relies on a base model, specifically a pre-trained BERT(Bidirectional Encoder Representations from Transformers) model adapter for Bengali language specifics. BERT, a state-of-the-art natural language processing model, utilizes neural network architecture to understand and represent contextual relationships within language data. The training process involves implementing appropriate loss functions, utilizing the AdamW optimizer for efficient weight updates during training, and incorporating gradient accumulation to accumulate gradients over multiple steps for stability. Mixed-precision training is employed to accelerate training without compromising precision.

The neural network implemented by BERT excels at capturing intricate language patterns, semantic nuances, and context dependencies, contributing significantly to BengaliBot's ability to comprehend user queries and generate meaningful responses. The neural network's inherent capacity for feature extraction and contextual representation empowers BengaliBot to navigate through diverse linguistic scenarios, providing an enhanced conversational experience.

The fine-tuning and iteration phase is a crucial aspect of the methodology, where the model is iteratively refined based on test results and user feedback. This adaptability ensures the model's continuous improvement and flexibility in handling various linguistic nuances. For better reactions over time, optional features are added, such as an adaptable learning system and bidirectional integration. BengaliBot is tested in the real world to make sure it works well in changing language

situations, and user feedback is gathered to make the model even better.

Our project intricately covers every facet of development, encompassing dataset preprocessing, model architecture, training, evaluation, testing and documentation. At this core the methodology emphasizes the integral role of neural networks, specifically exemplified by BERT model, in serving as the foundation for BengaliBot's natural language understanding, contextual conversation maintenance, and response generation capabilities.This comprehensive approach not only underscores the pivotal role of neural networks but also showcases their effectiveness in advancing the field of conversational AI for the Bengali language, delivering an interactive and efficient user experience.

V. Experimental Result

A lot of tests were done on BengaliBot to see how well it did in contextual question-answering situations. The tests focused on language understanding, contextual consistency, and response generation. The model was trained and tested using the Squad version 2.0 dataset, which was made better by translation. The following results show the most important parts of BengaliBot's work:

Accuracy and Precision:

BengaliBot did a great job of understanding user questions within the given context, with a success rate of over 90Precision measures, such as F1 score and exact match, showed how well the model could find and pull out important information from its surroundings.

Contextual Coherence:

During talks, BengaliBot kept contextual consistency by giving answers that make sense in the given situation. Understanding the subtleties of Bengali language patterns helped the context flow and made sure that exchanges made sense and made sense together.

Response Generation Quality:

Responses generated by BengaliBot exhibited a high-quality language output, demonstrating proficiency in natural language generation. The model effectively synthesized information from the context to formulate informative and contextually relevant answers.

User Interaction and Adaptability:

BengaliBot had talks with users and changed its answers based on what the users said and how they interacted with it. If optional learning methods were used, answers kept getting better over time, which made the system more adaptable.

User Feedback and Iterative Refinement:

Regular feedback from users was very important in improving BengaliBot's answers and fixing any problems that were found. The model was able to adapt to changing language subtleties thanks to iterative improvement based on user interactions.

Real-World Testing:

BengaliBot was tested in the real world by interacting with people in a variety of language situations that were meant to mimic real-life use. The model worked well in real-life situations, which showed that it could be used in active talking settings. In conclusion, these test results show that BengaliBot met the project's goals by creating a strong and flexible talking AI model for the Bengali language. The model is very good at understanding, responding in a logical way, and changing based on what the user does. This makes it a hopeful step forward in Bengali language processing.

Fig. 2. Test Result

Applications of Neural Network in this project:

Neural networks are crucial in the BengaliBot project since they are responsible for the fundamental operations, such as natural language comprehension, contextual conversation management, and generating responses. The project's core focuses on using recurrent neural networks (RNNs) and transformer architectures to understand and analyze Bengali text data. During the training phase, these neural networks undergo training using a carefully selected dataset that includes context, questions, and their related replies. BengaliBot is able to provide meaningful replies because it learns to identify complex verbal patterns, semantic subtleties, and contextual links. The models undergo fine-tuning to provide a minimum accuracy rate of 90 percent, highlighting the importance of neural network-based language models in reaching impressive performance standards. BengaliBot's use of neural networks enables it to proficiently negotiate intricate language structures, enhancing its efficacy in providing contextually precise and linguistically nuanced conversational interactions in the Bengali language.

VI. Limitations

While BengaliBot demonstrates advanced capabilities in contextual question-answering and conversational interactions, it also has certain limitations:

Context Dependency: BengaliBot heavily relies on the information present in the provided context. If a user's question goes beyond the scope of the context or requires external knowledge, the model may struggle to provide accurate or relevant answers.

Lack of Generalization: The model's training is based on a specific dataset, and its performance may be limited to the patterns and nuances present in that dataset. BengaliBot may face challenges in generalizing well to diverse linguistic scenarios or handling questions that deviate significantly from the training data.

Inability to Learn New Information in Real Time: BengaliBot can only learn from the training data; it can't get new information or adjust to changes that happen in real time. Because of this, it might not be able to answer questions about recent events or changes.

Sensitivity to Contextual Changes: Changes in the structure or format of the context may impact BengaliBot's performance. The model's sensitivity to contextual variations may result in suboptimal responses if the input format deviates significantly from its training data.

Translation Artifacts: Using translation application programming interfaces (APIs) to convert the Squad dataset to Bengali opens the door to the potential of developing translation artifacts. There is a possibility that the model's comprehension and response generation might be impacted by inaccuracies or subtleties that are lost during translation.

Difficulty with Creative or Abstract Queries: BengaliBot may face challenges in handling creative, abstract, or non-factual queries that require imaginative thinking. The model's design prioritizes factual information retrieval, and its responses may be less suitable for creative or hypothetical scenarios.

Understanding these limitations is crucial for managing user expectations and refining BengaliBot's capabilities through continuous updates and improvements.

VII. Conclusion

In conclusion, the BengaliBot project represents a remarkable advancement in Bengali conversational artificial intelligence. Using cutting edge neural networks like recurrent neural networks (RNNs) and transformers, BengaliBot is great at providing context-awareness, support for multiple languages, and flexibility through add-ons. The project acknowledges challenges such as sensitivity to context changes and creative queries, actively seeking continuous improvement. Real-world testing has confirmed BengaliBot's success in understanding the Bengali language, establishing its practical applicability. The integration of effective neural networks empowers BengaliBot to revolutionize user interactions and make significant contributions to the future of conversational agents in Bengali. With its focus on enhancing contextual understanding, accommodating multiple dialects, and providing optional features for user customization, BengaliBot is poised to transform the landscape of conversational AI in the Bengali language. This project serves as a testament to the potential and power of neural networks in creating intelligent, adaptable, and multilingual conversational companions. Finally, BengaliBot is more than just an AI model that can have conversations. It's a pledge to make technology play a bigger role in language diversity. As the project moves forward, its goal is to improve BengaliBot's features, listen to user comments, and look for new ways to make Bengali talking AI more useful. The trip goes on, with the goal of making Bengali talks easy and smart for a wide range of users.