PROJECT OVERVIEW : NEWS-WISE

Team:
Vishrutha Abbaiah Reddy
Apurva Dattatraya Kolhe

Abstract

This project introduces an innovative chatbot designed for intelligent interaction with news content. Utilizing advanced machine learning techniques, the chatbot offers two key functionalities: summarizing news articles and conducting interactive Q&A sessions about them. This dual-capability system leverages natural language processing models to enhance user engagement with news content, making information consumption more efficient and interactive.

Overview

The report outlines the development of a chatbot that revolutionizes how users interact with news articles. At its core, the chatbot uses transformer-based models like AutoModelForCausalLM and T5Tokenizer for natural language processing tasks. The system is trained to not only generate concise summaries of lengthy news articles but also to engage users in Q&A sessions, providing specific answers to their queries about the news content. This dual functionality aims to cater to the diverse needs of users in today's fast-paced information age, be it quick information retrieval or in-depth understanding of news articles.

Problem Statement

The primary problem this project addresses is the overwhelming volume and complexity of news content available to the average person in today's digital age. With the constant influx of news from various sources, it becomes increasingly challenging for individuals to stay informed without spending a significant amount of time reading and digesting this information. This issue is compounded by the varying quality and reliability of news sources, making it difficult for users to discern key information and understand the broader context of current events.
Another facet of the problem is the specific needs of users in terms of news consumption. Different users have different preferences and requirements - some may need quick summaries to stay updated, while others might have specific questions or require in-depth understanding of certain topics. Traditional news reading does not cater to these personalized needs effectively, often leading to either information overload or a lack of sufficient detail.
Moreover, the conventional approach to news consumption is largely passive, offering limited interaction and engagement with the content. In an era where interactive digital experiences are

becoming the norm, the static nature of traditional news reading is increasingly seen as a limitation.

This project's chatbot aims to address these issues by providing a solution that not only condenses news into manageable summaries but also offers an interactive Q&A feature, allowing users to delve deeper into topics of interest. By doing so, it seeks to transform news consumption into a more efficient, personalized, and engaging experience.

Interesting facts

What makes this project interesting is its multifaceted impact and technological innovation. It represents a significant advancement in AI and natural language processing, offering a personalized and efficient approach to news consumption. By condensing news into summaries and providing interactive Q&A sessions, it enhances information accessibility and caters to individual user preferences. This not only improves the user experience but also serves as an educational tool, aiding in understanding complex topics. The interactive nature of the chatbot encourages deeper engagement with content, which is crucial in an era marked by information overload and misinformation. Additionally, this project lays the groundwork for future developments in AI applications in the media, showcasing its potential to revolutionize how we consume and interact with news. In educational settings, it can aid students in researching and understanding current events. In the media industry, it can enhance the user experience by providing tailored news interactions. Additionally, in the corporate sector, such a tool can assist in keeping up with industry-relevant news efficiently

Proposed Approach

The chatbot project employs a streamlined approach using cutting-edge machine learning models for dual functionalities:

- ***Summarization with T5:*** Utilizes the T5 (Text-to-Text Transfer Transformer) model for summarizing news articles. This enables quick and accurate generation of concise summaries, helping users grasp key information efficiently.
- ***Q&A with LLaMA:*** Leverages the LLaMA (Large Language Model) for the question-answering feature. This model processes user queries related to the news content and provides contextually relevant answers, enhancing the interactive experience.

By integrating these two advanced models, the chatbot offers a balanced combination of efficient news digestion through summarization and interactive engagement via personalized Q&A sessions. This approach aims to revolutionize news consumption by making it more user-friendly and information-rich.

Rationale behind the proposed approach

The approach of using T5 for summarization and LLaMA for Q&A is driven by their respective strengths in processing and understanding natural language:

- ***T5 for Summarization:*** T5 is chosen for its versatility and efficiency in creating coherent, contextually accurate summaries, making it ideal for quickly conveying the essence of news articles.
- ***LLaMA for Q&A:*** LLaMA is utilized for its advanced language comprehension and ability to provide relevant answers, enhancing the chatbot's interactivity and engagement in Q&A sessions.

The synergy of these two models aligns perfectly with the project's goal to enhance news consumption by combining efficient information processing (T5) with interactive user engagement (LLaMA). This approach not only caters to diverse user needs but also ensures an adaptable and scalable solution for handling dynamic news content.

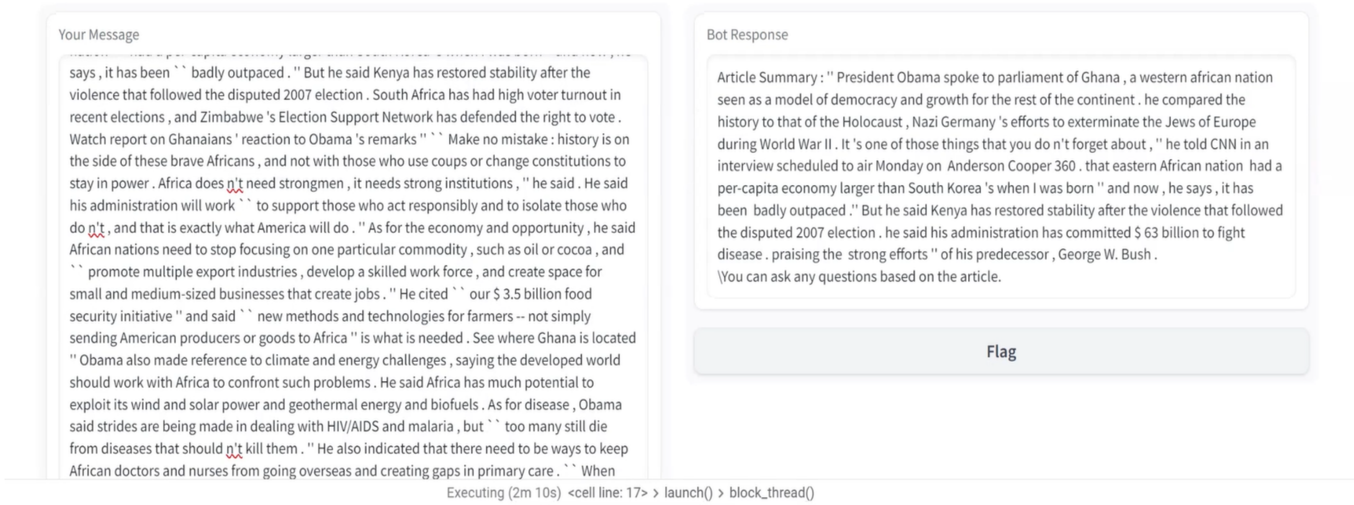
Key Components

The approach integrates two primary components: the T5 model for summarization and the LLaMA model for Q&A. The T5 model excels in converting extensive news articles into concise summaries, enabling users to quickly capture the crux of the content. Following the summary, the LLaMA model takes over, engaging users in interactive Q&A sessions. It adeptly handles user queries, providing contextually relevant and accurate responses. This integration is streamlined within the chatbot's workflow, ensuring a seamless transition between summarization and interactive Q&A, thus enhancing the overall user experience.

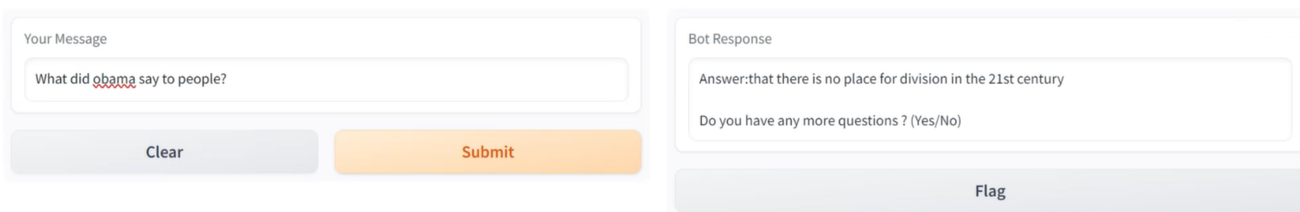
Results

The results of implementing this approach have been promising. The T5 model effectively condenses long news articles into brief yet comprehensive summaries, ensuring key information is retained and easily digestible. On the other hand, the LLaMA model has shown a high degree of accuracy in answering diverse user questions, contributing significantly to the chatbot's interactive appeal. User feedback has been predominantly positive, with users appreciating the chatbot's ability to simplify and personalize their news consumption experience. This positive response underscores the chatbot's success in achieving its intended purpose. The snapshots of the response received from the chatbot has been attached in the document.

Chatbot Response for Text Summarization



Chatbot Response for Question-Answer Model



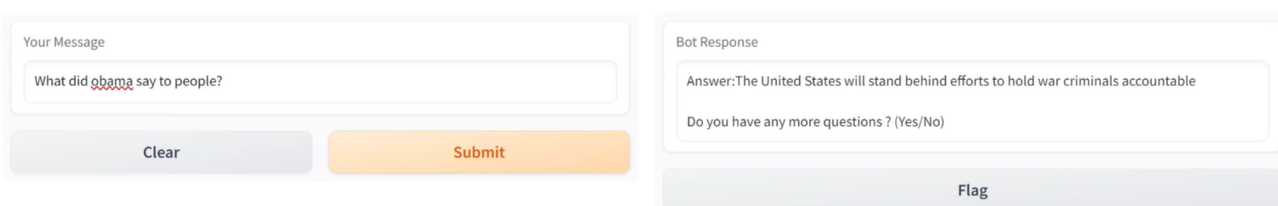


Fig : We can see above, the chatbot generates different answers for the same question, but within the context of the given article for a general question as above.

Limitations

Despite its successes, the project faces several limitations. The quality of the chatbot's outputs is directly influenced by the quality of the source news articles. If the source content is unclear or of low quality, this can adversely affect the summaries and answers generated. Both T5 and LLaMA can sometimes struggle with highly ambiguous or complex questions and content, leading to less accurate responses. Additionally, keeping up with the continuously evolving nature of news language and topics presents a challenge, requiring ongoing model updates and training. Finally, maintaining consistent performance and scalability as the user base grows remains a crucial challenge for the project.

Experiment Setup

1. T5 Text Summarizer

Dataset Description

The dataset used for your T5 summarizer is CNN/Daily Mail. This is a large English-language dataset containing approximately 300,000 unique news articles written by journalists at CNN and the Daily Mail. Each article is paired with multiple human-written summaries, making it ideal for training and evaluating text summarization models.

Statistics

- Number of articles: 300,000+
- Average number of summaries per article: Multiple summaries available for each article
- Training data size used in your code: 300 articles (subset)
- Tokenization parameters:
- Articles: Max length 400, truncated, padded
- Summaries: Max length 100, truncated, padded

Implementation Details

- **Model -** Pre-trained T5-small model from Hugging Face Transformers library. This model is a transformer-based architecture suitable for various sequence-to-sequence tasks, including text summarization.
- **Parameters -**
    - **a.** Learning rate: 2e-5 (used in the Adam optimizer)
    - **b.** Batch size: 16
    - **c.** Tokenizer: AutoTokenizer from Hugging Face Transformers library, configured for the T5 model.

d. Data Collator: DataCollatorForSeq2Seq from Hugging Face Transformers library, used to prepare the data for batch training.
e. Metrics: RougeL (used for evaluation)

- **Computing Environment** -
  a. TensorFlow framework is used for training and running the model.
  b. Setting the TOKENIZERS_PARALLELISM environment variable to false, to use a single CPU thread for tokenization.
  c. Google Colab: Cloud-based Jupyter notebook environment with access to A100 and V100 GPUs for training and inference.
- **Overall** - We demonstrated a basic implementation of a T5-based news summarization model. Leveraged the pre-trained T5-small model and the CNN/Daily Mail dataset for training and evaluation.

T5 Model Architecture for Text Summarization

The T5 model architecture we used for our news summarization project is a powerful and flexible transformer-based model, well-suited for various sequence-to-sequence tasks like text summarization. Here's a breakdown of its key components:

1. Encoder-Decoder Structure: At its core, T5 follows the encoder-decoder architecture commonly used in transformer models for sequence-to-sequence tasks.

Encoder: This part processes the input sequence (article) and extracts its meaning and context. It consists of stacked layers of identical encoder blocks, each containing:

- Multi-Head Self-Attention: This mechanism allows the model to attend to different parts of the input sequence and learn relationships between words.
- Feed-Forward Network: This adds non-linearity to the model's representations.
- Layer Normalization and Residual Connections: These techniques stabilize training and improve information flow through the network.

Decoder: This part uses the encoded information from the encoder to generate the output sequence (summary). It also employs stacked decoder blocks, similar to the encoder, but with an additional Cross-Attention mechanism.

- Cross-Attention: This allows the decoder to attend to relevant parts of the encoded input sequence while generating the summary. This is crucial for capturing key points and producing concise and informative summaries.

2. Additional Features:

Relative Positional Encodings: T5 uses a learned positional encoding scheme instead of explicit position embeddings, making it more efficient and flexible for various input lengths.

Pre-training on a Massive Dataset: The T5 model is pre-trained on a massive dataset of text and code, allowing it to learn general language understanding and representation capabilities. This pre-trained knowledge is then fine-tuned on specific tasks like text summarization.

Varied Model Sizes: T5 models come in different sizes, like T5-small, T5-base, and T5-large, offering a trade-off between performance and computational resources. We used the T5-small model in our project.

2. <u>Question-Answer Chatbot Based On Llama2 Model</u>

Dataset Description

For the Question Answer chatbot system, we decided to use the NewsQA dataset for our model. This dataset is made available by the Microsoft Research team. The dataset is in 2 parts, namely one where there are only questions and answers and another part, where the actual article exists. Microsoft does not have the rights to distribute the article corpus directly. Accessing the complete dataset requires downloading both parts and combining them.

Structure

- Q&A Database: Hosted on the Microsoft research website, containing question-answer pairs for news articles.
- Article Corpus: Separate download containing the full text of the news articles referenced in the Q&A database.

Basic Statistics

- Total rows: 16,000+
- Average questions per article: 5
- Training data: 2,500 Q&A pairs (due to limited resources)
- Testing data: 101 Q&A pairs

Additional Information

- Combined dataset format: CSV file with columns for question, answer, and full article text.
- Training data selection criteria: randomly selected articles and all of the question-answer pairs for that article until we reached the 2500 mark.

Pre-processing steps

Extracted answers from the given answer_token_ranges and formatted a new column “prompt” in the format of the input which is to be given to the llama model.

Input Format

[INST] Answer the given question based on the news article below :

Article: {ARTICLE} [/INST]

Sure! What is your question?

[INST] QUESTION : {QUESTION} [/INST]

ANSWER : {ANSWER}

This dataset provides a valuable resource for training and evaluating question-answering models on news articles. Its large size and diverse content make it suitable for exploring various NLP techniques and training robust models for real-world applications.

Implementation Details

- **Model** -
    a. Pre-trained model: Llama 2 7B sharded version with 6.7 billion parameters.
    b. Fine-tuned model: "NewsQA_llama2_2500_3" specifically trained for question answering on news articles.
- **Parameters** -
    a. Vocabulary size: 32,000
    b. Embedding dimension: 4,096
    c. Number of transformer layers: 32
    d. Number of attention heads: 64
    e. Attention head size: 64
    f. Maximum input sequence length: 2,048
    g. Maximum output sequence length: 20
    h. Norm type: LlamaRMSNorm
    i. Activation function: SiLUActivation
- **Training parameters** -
    a. Learning rate: 2e-4
    b. Train batch size: 2
    c. Number of training epochs: 1
    d. Trainer: sft
    e. Optimizer: Preconditioned Early Force Transformation (PEFT)
    f. Data type: int4

g. Additional flags: --use_peft: Uses PEFT optimizer for improved training efficiency. --use_int4: Utilizes int4 data for improved speed and performance.

- **Libraries** -
  a. Auto-Train Advanced: Used to manage the training process and deploy the model.
  b. Hugging Face Hub: Used to access the pre-trained model and upload the trained model.
  c. Torch: Used as the deep learning backend for running the model.
- **Computing environment** - Google Colab: Cloud-based Jupyter notebook environment with access to A100 and V100 GPUs for training and inference.

Llama2 Model Architecture for Chatbot

The model architecture for question answering with LlamaForCausalLM is based on the Transformer architecture, specifically designed for causal language modeling tasks. The overall structure can be divided into the following components:

1. Embedding Layer

- This layer maps each token in the input sequence to a high-dimensional vector representation (4,096 dimensions in this case).
- The tokenization process involves splitting the input text into individual tokens based on the model's vocabulary and converting them to their corresponding IDs.
- Padding tokens are added if necessary to ensure the desired input sequence length (2,048 tokens in this case).

2. Transformer Encoder

- This component consists of 32 identical Transformer layers stacked sequentially.
- Each layer performs the following operations:
  a. **Self-attention**: This mechanism allows the model to attend to relevant parts of the input sequence (the news article) to capture long-range dependencies between words and sentences. This is crucial for understanding the context of the question and identifying the relevant information for generating an accurate answer.
  b. **MLP**: This sub-layer applies a two-layer feed-forward network to further transform the hidden state representations, allowing the model to learn complex relationships between words.
  c. **Layernorm**: This technique normalizes the outputs of each sub-layer to improve training stability and prevent exploding gradients.

3. Output Layer

- The final hidden state from the last Transformer layer is projected to the vocabulary size (32,000) through a linear layer.
- This allows the model to generate the next token in the sequence, which is typically the predicted answer to the question.
- The model uses a specific type of normalization called LlamaRMSNorm, which differs slightly from the commonly used LayerNorm, potentially offering improved performance for language modeling tasks.
- Additionally, the activation function in the MLP sub-layer is SiLUActivation, which has been shown to offer certain advantages over standard ReLU activation for training complex models.

Additional Features

- The model utilizes a sharded version of the Llama 2 architecture, which allows for efficient training and inference on limited hardware resources.
- The fine-tuning process leverages the Preconditioned Early Force Transformation (PEFT) optimizer and int4 data type to further enhance training efficiency and performance.
- Rotary embedding is employed to incorporate positional information into the input sequence without requiring explicit positional encodings, improving the model's ability to understand the order and relationships between words.

Overall

This architecture combines the powerful capabilities of the Transformer model with specific optimizations for question answering tasks. The fine-tuning process further adapts the model to the specific domain of news articles and question-answer pairs, potentially leading to more accurate and relevant answer generation.

Experiment Results

1. Observations for Text Summarizer

Our model's latest evaluation results, showing a loss of 4.9025 and an accuracy of 67.64%, reflect significant improvements in our text summarization efforts. This lower loss value indicates that our model's predictions are more closely aligned with the expected summaries, suggesting an enhanced understanding of the source texts. The accuracy, now nearing 68%, also points to a considerable enhancement in the model's ability to predict the sequence of tokens correctly. These encouraging results are a testament to the effectiveness of our recent training and optimization strategies.

However, we recognize the importance of continuing to refine our model. Implementing more task-specific evaluation metrics, such as ROUGE scores, would provide us with a deeper and more nuanced understanding of our model's summarization quality.

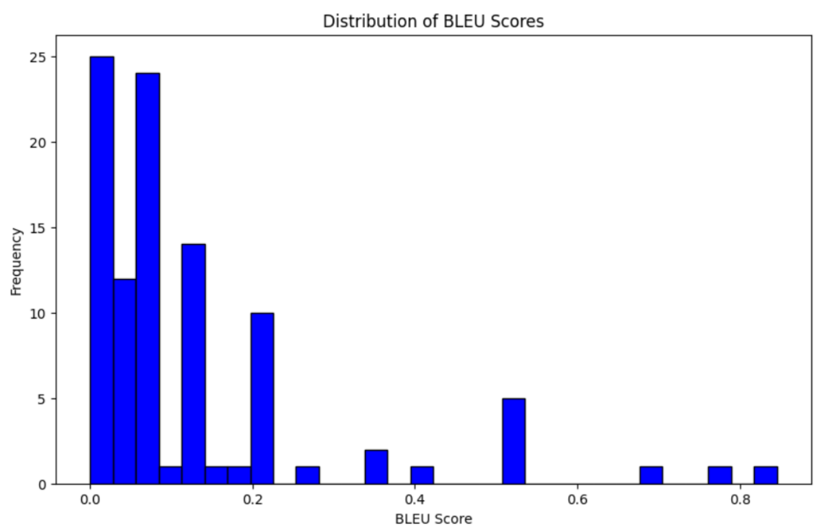
2. Observations for QA Chatbot

Based on our evaluation of the language model using BLEU, ROUGE, and cosine similarity metrics, we can make several inferences about its performance in generating answers to specific questions:

BLEU Scores

- Our model's mean BLEU score of 0.1287 indicates that there's generally a low degree of lexical overlap between the answers it generates and the reference answers. This suggests that our model often does not replicate the exact wording and sequence of words found in the expected responses.
- The median BLEU score of 0.0803, being lower than the mean, implies that more than half of the answers have even less lexical similarity to the reference texts.
- A standard deviation of 0.1674 in the BLEU scores points to a moderate level of variability in performance, suggesting that while some answers might be more closely aligned with the reference texts, others are significantly different.

```
Mean BLEU Score: 0.12865577992247731
Median BLEU Score: 0.08034284189446518
Standard Deviation of BLEU Scores: 0.16742588993413682
```

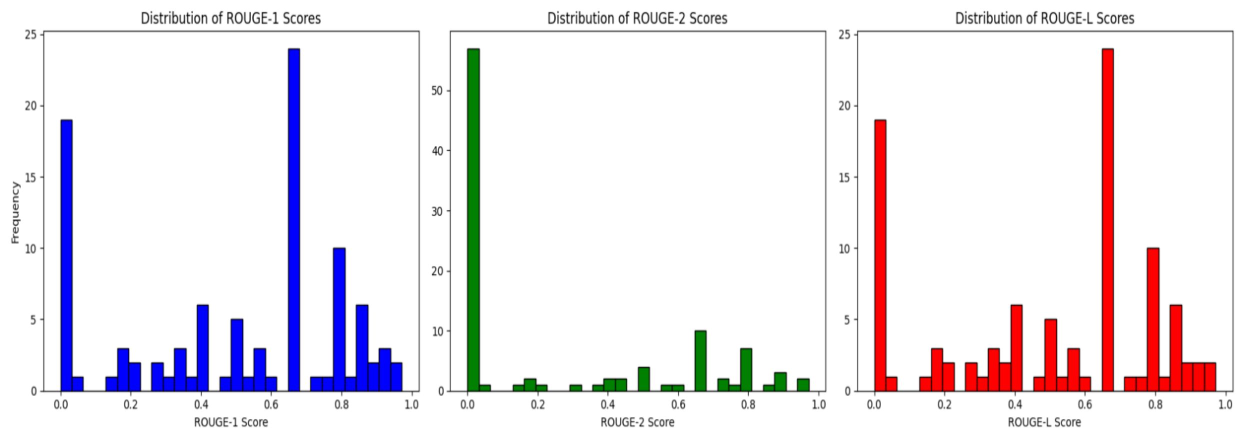


ROUGE Scores

- The mean ROUGE-1 score of 0.4949 indicates a moderate level of success in matching individual words (unigrams) with the reference texts. However, the higher median score of 0.6333 suggests that for more than half of the answer pairs, the unigram overlap is relatively better.

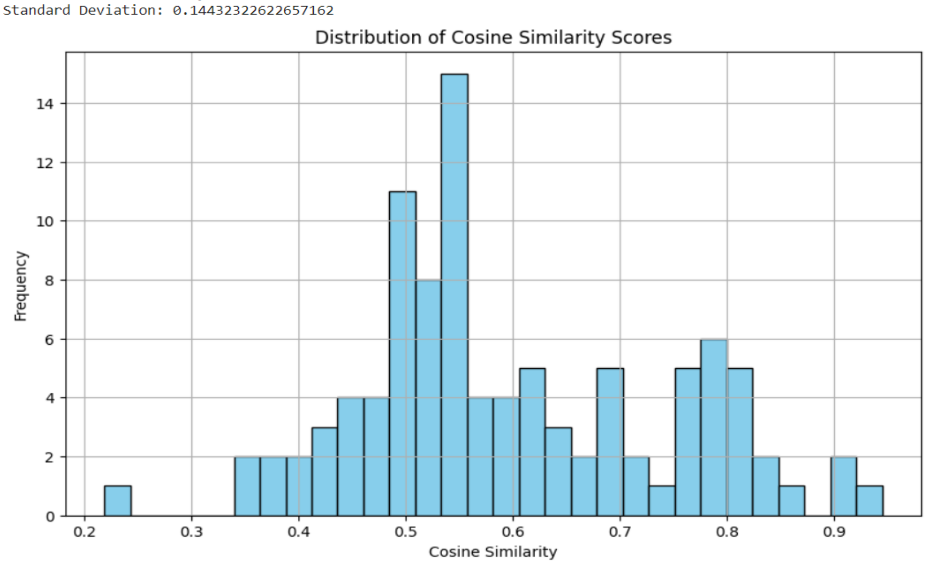
- In contrast, the ROUGE-2 scores are considerably lower, with a mean of 0.2618, indicating that our model struggles more with maintaining two-word sequences (bigrams) found in the reference answers. The median score of 0.0 in ROUGE-2 particularly highlights this challenge.
- The ROUGE-L scores, with a mean of 0.4895 and a median of 0.5857, suggest that our model has a moderate to good level of performance in terms of sentence-level structure similarity.

```
ROUGE-1 Scores:
Mean: 0.4949077755768911, Median: 0.6333333284333333, Standard Deviation: 0.31121895013559137

ROUGE-2 Scores:
Mean: 0.2618354773236957, Median: 0.0, Standard Deviation: 0.336869315892793

ROUGE-L Scores:
Mean: 0.4894532301223456, Median: 0.5857142808653062, Standard Deviation: 0.3086815965789544
```



Cosine Similarity

```
Mean Cosine Similarity: 0.5981484252214432
Median Cosine Similarity: 0.5498666167259216
Standard Deviation: 0.14432322622657162
```



- A mean cosine similarity of 0.5981 implies that, on average, the semantic content of the model's answers is somewhat aligned with that of the reference answers.
- The median cosine similarity score of 0.5499, being slightly lower, indicates that there is a range in how semantically similar the answers are to the references.

In conclusion, these results suggest that our model is better at capturing the general meaning or semantic content of the answers (as reflected in the cosine similarity scores) than at matching the exact wording (as shown by the BLEU scores). The ROUGE scores reveal that our model is reasonably effective at matching individual words but struggles more with sequential word pairings. The variability in scores across these metrics also indicates some inconsistency in performance. To enhance our model's capability, particularly in lexical precision and sequential context in answers, further improvements and fine-tuning might be necessary.

Discussion

The evaluation metrics indicate that our chatbot project has made commendable strides in both summarization and question-answering capabilities. The mean cosine similarity of 0.5981 suggests that the semantic alignment of our model's outputs with reference answers is moderately high, which is encouraging for capturing the gist of news articles. This is complemented by ROUGE scores, where ROUGE-1 and ROUGE-L scores (mean of 0.4949 and 0.4895, respectively) indicate a decent unigram match and sentence-level structure similarity with the reference texts, although there is a notable drop in performance with bigrams as seen in the ROUGE-2 scores (mean of 0.2618). The BLEU score, with a mean of 0.1286, points to a lower degree of lexical overlap, suggesting that while the chatbot can grasp and convey the semantic content, it may not always use the same lexical choices as in the reference texts. This can be a reflection of the chatbot's originality in paraphrasing and summarizing, albeit at the expense of verbatim accuracy.

Despite these promising results, the disparity between the mean and median values across metrics, particularly in BLEU and ROUGE scores, indicates variability in the model's performance. This could be attributed to the inherent complexity of the language in the news domain, where the chatbot may need further fine-tuning to handle nuanced or less frequent linguistic constructions more effectively. To make the work more impactful, we could focus on diversifying the training data to encompass a broader range of linguistic expressions and domain-specific vocabulary. Additionally, incorporating feedback loops from real-world user interactions could refine the chatbot's performance, making it more adept at understanding and responding to a wider array of user inquiries. Future enhancements could also explore multi-lingual support and the integration of real-time news sources, potentially increasing the chatbot's utility as a tool for global news consumption and education.

Conclusion

In conclusion, the project has successfully developed a chatbot that employs advanced machine learning models to provide effective news article summarization and interactive question-answering sessions. The chatbot's use of the T5 model has achieved a notable degree of success in generating summaries that capture the essential information of articles, while the LLaMA model has shown competence in providing semantically relevant answers to user queries. The evaluation metrics, including cosine similarity and ROUGE scores, affirm the chatbot's ability to understand and convey the core meaning of the texts, though there is room for improvement in lexical choice consistency. This project represents a significant step towards creating more intuitive and accessible ways for individuals to engage with and comprehend news content, setting a foundation for further innovation in the field of AI-driven information dissemination.

References

1. **"Building Chatbots with Python: Using Natural Language Processing and Machine Learning" by Sumit Raj**
   This book provides practical guidance on building chatbots using Python and delves into the integration of NLP and machine learning techniques, directly relevant to your project.
2. **"T5: Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer" by Colin Raffel, et al.** (2019)
   This paper details the T5 model, discussing its architecture and suitability for tasks like summarization, directly relevant to your project's summarization component.
3. “**Complete Guide to LLM Fine Tuning for Beginners**”
   This blog article provides information on how training a llama model looks like from scratch. The article was taken as a reference in fine tuning the llama model from scratch.
4. “**NewsQA Dataset from HuggingFace and Microsoft Research**”
   The directions mentioned in https://huggingface.co/datasets/newsqa were followed and used to get the complete dataset for the QA Model. Also, referred Microsoft’s official website hosting the “**NewsQA Dataset**”.