News Summarization using T5 Transformer

Kunal Inglunkar Individual report

Introduction

News summarization is essential in a world inundated with information. It saves time, helps manage information overload, and ensures accessibility for a diverse audience. With fast-paced news cycles, summarization allows quick access to key insights, catering to the needs of professionals, researchers, and the general public alike. In a nutshell, it provides efficiency, relevance, and timely updates in our information-rich environment.

In this project, we leverage the power of the T5 (Text-To-Text Transfer Transformer) model, a transformer-based architecture known for its effectiveness in various NLP tasks. The model is pre-trained on a diverse range of tasks and fine-tuned specifically for abstractive summarization. Our experiments are conducted using the Multi-News dataset, a comprehensive collection of news articles paired with human-generated summaries, providing a robust benchmark for evaluating summarization models.

Background information on the development of the algorithm

Introduction to T5 Model:

The T5 model, developed by Google Research, stands as a formidable transformer-based architecture within the realm of Natural Language Processing (NLP). A successor to influential models like BERT and GPT, T5 excels across a spectrum of NLP tasks, showcasing its prowess in text summarization.

Transformer Architecture:

Originating from the seminal work "Attention Is All You Need" by Vaswani et al., the transformer architecture revolutionized NLP by eliminating the need for recurrent or convolutional layers. T5 builds upon this foundation, leveraging self-attention mechanisms for efficient information processing.

Core Components:

The T5 model employs an encoder-decoder structure, with the encoder handling input text and the decoder generating output summaries. Key to its efficiency is the self-attention mechanism, allowing differential weighting of input sequence elements for effective capturing of long-range dependencies. Multi-head attention, a parallelized version of self-attention, further enhances the model's ability to focus on distinct aspects simultaneously.

T5-Specific Features:

T5 introduces the innovative concept of "Text-To-Text Transfer," framing all NLP tasks as transforming input text into output text. This unified approach simplifies training and fine-tuning processes for diverse tasks, including summarization.

Text Summarization with T5:

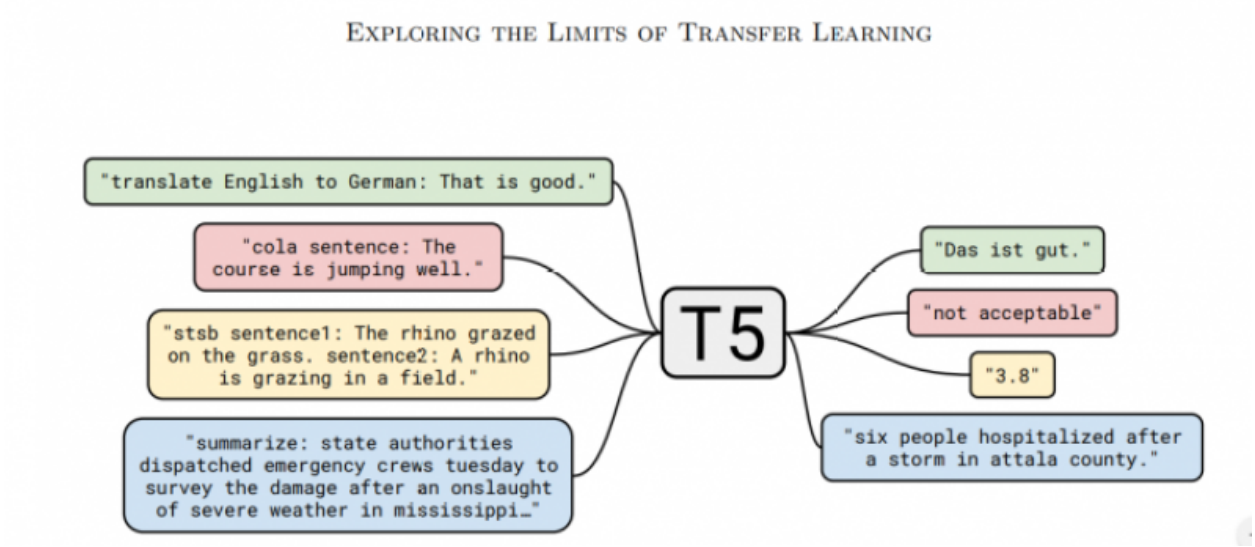
A highlight of T5's capabilities lies in text summarization. By prefixing the input text with "summarize:", T5 can succinctly capture the essence of lengthy documents, making it particularly valuable for applications like news articles, scientific papers, and legal documents.

Unified NLU and NLG:

T5 unifies Natural Language Understanding (NLU) and Natural Language Generation (NLG) tasks through a sequence-to-sequence approach in the encoder-decoder variant. For text classification, the model uses the text as the encoder input, with the decoder generating the label as natural text rather than a class.

Training Methodology:

The T5 architecture, the original Transformer variant, undergoes training on the large crawled C4 dataset. Masked language modeling is employed as a training method, with the largest T5 model boasting an impressive 11 billion parameters and achieving State-of-the-Art (SOTA) results across various benchmarks.



Description of Individual Work

Introduction:
In the pursuit of effective news summarization, this project I began by leveraging the Hugging Face load_dataset function to acquire the Multi-News dataset. The dataset encompasses news articles paired with corresponding summaries, serving as a valuable resource for natural language processing (NLP) tasks. The initial focus was on constructing an LSTM model with an encoder-decoder architecture for summarization. However, early implementation and training stages revealed challenges, prompting a strategic shift towards a transformer-based approach.

Data Preprocessing:
A crucial step in preparing the data involved comprehensive preprocessing. This encompassed the application of regular expression removal, stop-word elimination, stemming, and tokenization. Further, the text was embedded using Word2Vec, a technique that enhances the representation of words within the dataset. These preprocessing steps aimed to optimize the input for subsequent model training.

Model Transition to Transformer:
Recognizing the limitations of the LSTM model, the project transitioned to a transformer-based model, specifically the Text-To-Text Transfer Transformer (T5). The decision to pivot to T5 was rooted in its proven effectiveness for text summarization tasks. This transformer architecture promised improved performance and scalability, addressing the encountered challenges in the initial LSTM model.

Hyperparameter Tuning:
Fine-tuning of the transformer model involved an exploration of various hyperparameters to optimize performance. This meticulous process aimed to enhance the model's ability to generate coherent and accurate summaries. The primary evaluation metric employed was the ROUGE score, providing insights into the model's summarization quality under different configurations.

Web Application Development:
Upon successful model development, the next step involved creating a user-friendly application for practical use. A basic Streamlit web app was designed to allow users to input text for summarization. The app seamlessly integrated the trained transformer model, generating concise and meaningful summaries in response to user-provided content.

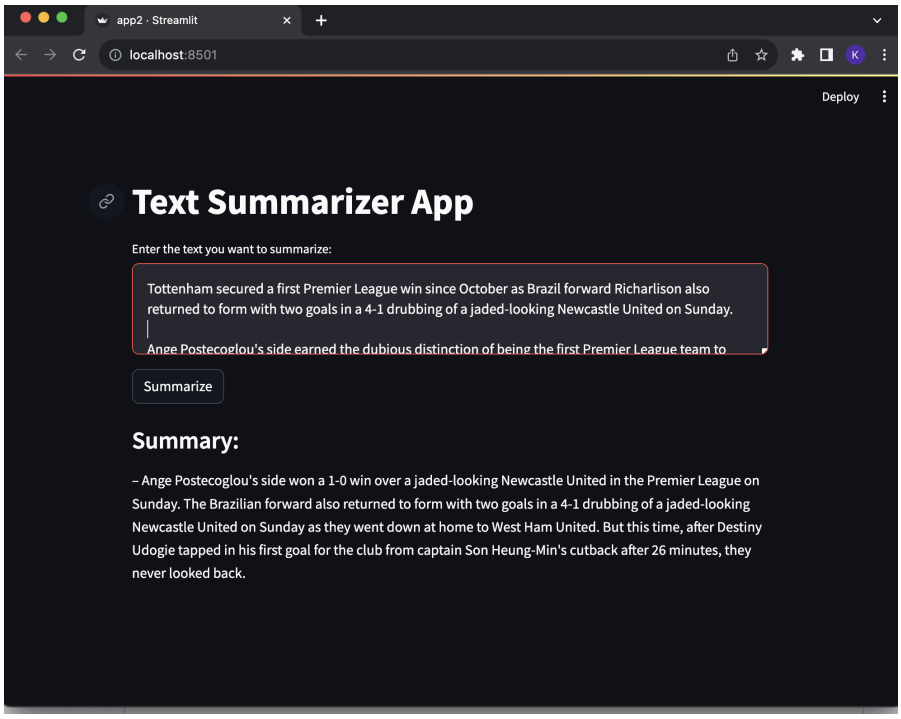
Results

To test our model we copied an article from ESPN(https://www.espn.com/soccer/report/_/gameId/671189) of about 552 word and then generated summary using quillbot summarize provide reference and see ROUGE scores evaluating the quality of the summary.
The Summary provided by quillnot is below

"Tottenham secured their first Premier League win since October, with Brazil forward Richarlison scoring two goals in a 4-1 win against Newcastle United. The team, led by Angel Postecoglou, became the first Premier League team to lead 1-0 in five successive games without winning any of them in midweek. The win left Tottenham in fifth place with 30 points from 16 games, seven behind leaders Liverpool in a compelling title race. Newcastle, however, fell to seventh place with 26 points. The win was a result of a strong performance by Tottenham, who had only collected one point in their last five games. Postecoglou praised the team's confidence and the positive energy of their players. The win leaves Newcastle in seventh place with 26 points. The match was a testament to the team's resilience and determination."

The generated summary from out model is :

Generated summary using T5:



- The generated summary provides a brief overview of the key points in the input text. However, it seems to have some issues, such as mentioning a 3-0 win over Everton,

which may be a mistake or confusion in the model's output. Additionally, it refers to a 1-0 win over West Ham United, which is inconsistent with the details in the input text. These discrepancies suggest that the model may not have accurately captured the information from the original passage.

- The ROUGE scores provided are metrics that evaluate the quality of a generated summary by comparing it to a reference summary. Here's an explanation of the ROUGE scores:

  rouge-1:

  - Recall (r): 0.36
  - Precision (p): 0.53
  - F1-Score (f): 0.43

  Explanation: For unigram (single-word) overlap between the generated summary and the reference summary, the recall is 0.36, indicating that 36% of the reference summary's unigrams are present in the generated summary. The precision is 0.53, suggesting that 53% of the unigrams in the generated summary are relevant. The F1-score, which is the harmonic mean of precision and recall, is 0.43.

  rouge-2:

  - Recall (r): 0.12
  - Precision (p): 0.2
  - F1-Score (f): 0.15

  Explanation: This metric considers bigram (two-word) overlap. The recall of 0.12 indicates that only 12% of the reference summary's bigrams are present in the generated summary, while the precision of 0.2 suggests that 20% of the bigrams in the generated summary are relevant. The F1-score for bigrams is 0.15.

  rouge-l:

  - Recall (r): 0.33
  - Precision (p): 0.48
  - F1-Score (f): 0.39

  Explanation: Rouge-l measures the overlap in longest common subsequences, which is more flexible than strict word overlap. The recall of 0.33 means that 33% of the longest common subsequences in the reference summary are present in the generated summary.

The precision of 0.48 indicates that 48% of the longest common subsequences in the generated summary are relevant. The F1-score for rouge-l is 0.39.

In summary, these ROUGE scores suggest that while there is some overlap between the generated summary and the reference summary, there is room for improvement, particularly in capturing more of the reference summary's content (higher recall) while maintaining precision. The F1-scores provide a balanced view of the trade-off between precision and recall. Adjustments to the summarization model or fine-tuning may be considered to enhance its performance**.**

Summary and conclusions

In conclusion for the project I preprocessed data and tried building a lstm model which didn't work hence tried fine tuning T5 transformer. And used this model to get ROUGE scores and build a streamlit app one for getting ROUGE score which uses reference summary and another where user just provide the link to the article and the app summarize the article

percentage of the code used from other source: 20%

Reference:

https://medium.com/@ajazturki10/text-summarization-with-t5-pytorch-and-pytorch-lightning-b7a319ec9ea2#:~:text=T5%2C%20or%20Text%2Dto%2D,machine%20translation%2C%20and%20sentiment%20classification.

https://pythonsimplified.com/build-a-text-summarization-app-using-streamlit-in-30-minutes/