ABSTRACTIVE TEXT SUMMARIZATION USING BART

P. Gayathri 1, Ch. Dhana Lakshmi 2, M. Naga Sirisha 3 and Sd. Rizwana 4

1,2,3 student of Department of Computer Science and Engineering, Narasaraopet Engineering College

4 Faculty of Department of Computer Science and Engineering, Narasaraopet Engineering College, Narasaraopet

Email: gayathripalepu660@gmail.com , dhanalakshmichelli08@gmail.com, mannepuli123@gmail.com, syedrizwananrt@gmail.com

Abstract

We have witnessed the rise of automation in recent years for human convenience. With the use of ML learning, we get closer to realizing a general in nature AI. Natural Language Processing, Computer Vision, and Machine Learning are the three main subfields of artificial intelligence. Automated Text Summarization is a key component of Natural Language Processing, which entails the comprehension and manipulation of human language. Text summarizing is reducing a long document to a concise summary. While keeping the information's context (or meaning), it produces information that is fluid and coherent. Making a manual summary is a challenging process for humans because it necessitates a careful examination of the full document.

Keywords: Machine Learning, Text Summarization, BART Model.

1. Introduction

Text summarizing, which involves shrinking the original document's size while retaining its original information, produces a summary that is less than half the size of the original document's main text. One could think about summarization as a two-step procedure. The first stage is to extract important ideas or sequences from the original text by creating an intermediary vector or file. This step may also involve any text pre-processing, such as tokenization, tagging, or other operations, that is necessary. This intermediate file is used to create a summary in the following phase. One example of a text summarizer that enables users to find the news that most interests them is News Blaster. The provocation of writing a concise, pinpoint, and synopsis of a lengthier text document is known as text summarization. In order to better assist in the discovery of relevant information and to consume relevant information more quickly, approaches for automatic text summarizing are urgently needed. Using self-attentions, this technique permits interactions not just between words but also between phrases and words.

The major goal of this project is to lessen the emphasis on giving a trustworthy summary of a lengthy text, which mostly saves people's important time. so that they can concentrate on their best projects. We are employing the BART model of abstractive text summarization to condense this lengthy paper[6]. It greatly aids pupils in learning how to identify key concepts and track down relevant information to bolster those concepts and make them more useful. It aids the kids in developing better concentration abilities so they can concentrate on words and phrases from the lengthy material that has been supplied.

The dataset we have used is CNN Daily Mail Dataset, which contains 3 attributes as id, article, highlights. The id shows the unique id's for each and every article present in the dataset. And the article attribute contains the long document or article which we need to summarize. And finally highlights attribute contains the summary of the article which can used further in calculating the rouge score passing as reference parameter.

2. Abstractive Text Summarization

Abstractive text summarization is one of the type of text summarization. It evolve newly discovered words and sentences, integrate those in a significant method and then attach the nearly all pivotal particulars in the original document. As a result, abstractive approaches are added arduous to use than extractive summarization strategy and cost more to compute.

As human language is sequential in nature, it has been discovered that RNNs and models based on them perform better than alternative methods. One of the fundamental models for abstractive text summarization is sequence to sequence[5]. One sequence is followed by another (e.g. machine translation). It does this by utilizing DNNs. For extended carry of sequences and to prevent the back propagation issue, this method especially leverages LSTM. The input of one cell becomes the output of the next, and so forth. This enables it to understand the order through a sentence. Encoder-decoder architecture is usually used. The encoder converts an input into a invisible vector that correlate with to it and repress both the item and its value. This concealed vector is transmitted.

3. BART Model

BART is a denoising autoencoder that was trained as a sequence-to-sequence model. This implies that a refined BART model can accept one text sequence as input and output another text sequence. This kind of approach is applicable to text summarization [2], question answering, machine rendering question answering on a specific corpus, question answering, and sequence categorization (labeling input text sentences or tokens). Sentence evocation is other assignment that evaluates whether two or more sentences are reasonably expansion of one another or wisely connected to a given submission.

3.1 Architecture

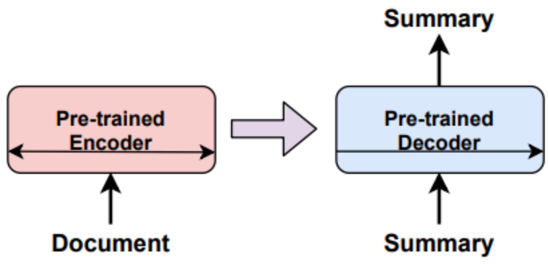

Figure 1: Architecture of BART Model

The building design is very alike to that of Bidirectional Encoder Representation from Transformers, with the backing exceptions: (1) Every part of the decoder enact cross-attention above the last invisible part of the encoder in addition (as in the transformer sequence-to-sequence model); and (2) Unlike BART[7], Bidirectional Encoder Representation from Transformers employs totalling feed-forward network prior to word prediction. Bidirectional Auto-Regressive Transformers has about ten percent more limitation overall than the Bidirectional Encoder Representation from Transformers model of the same size.

3.2 Pre-training of BART

The cross-entropy connecting the production of the decoder and the real document—the remodeling debt is optimized after manipulated documents in order to train BART [1]. BART enables us to apply any kind of document corruption, unlike existing denoising autoencoders, which are tailored to precise noising plans. Bidirectional Auto-Regressive Transformers is similar to a language model in the inferior case, where all source-associated information is damaged. We test a number of recently developed and creative transformations, but we think there is a lot of room for the creation of additional fresh options. Examples of the transformations we utilized are displayed in Figure 2, and a summary of them is given below.

Token Masking:

Token Masking is used to restore any independent tokens in a sentence. Established the remaining of the succession, the model evolves the potential to forecast the only one token**.**

Sentence Permutation:

Arbitrary permutations are used to whole stops-separated sentences. This support the model's studying of how sentences are wisely implicit.

Document Rotation:

The sequence of the document is interchange to begin with a chosen token[2]. At the edge of the document, the data that came before the token is appended. This supplies particulars on the extensive organization of the document as well as what a document's starting or closing view like.

Token Deletion:

Tokens are eliminated at arbitrary. The model require to have the potential to forecast token data and recognize the position where a token was removed from.

Text Infilling:

Text Infilling places a masked token into a arbitrary chosen position or substitutes word sequences with a only one mask token. Almost precise method is aggregate of sentence permutation and text infilling.

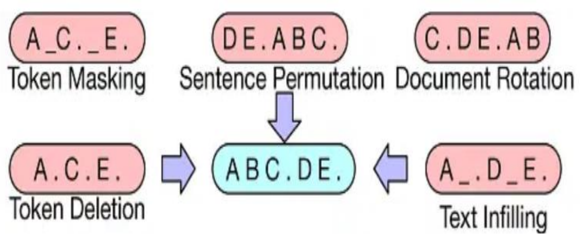

Figure 2: Pre-training Techniques

4. Dataset

The dataset we have used is CNN Daily Mail Dataset, Which contains plenty of news articles int it.

	id	article	highlights
0	0001d1afc246a7964130f43ae940af6bc6c57f01	By . Associated Press . PUBLISHED: . 14:11 EST...	Bishop John Folda, of North Dakota, is taking ...
1	0002095e55fcbd3a2f366d9bf92a95433dc305ef	(CNN) -- Ralph Mata was an internal affairs li...	Criminal complaint: Cop used his role to help ...
2	00027e965c8264c35cc1bc55556db388da82b07f	A drunk driver who killed a young woman in a h...	Craig Eccleston-Todd, 27, had drunk at least t...
3	0002c17436637c4fe1837c935c04de47adb18e9a	(CNN) -- With a breezy sweep of his pen Presid...	Nina dos Santos says Europe must be ready to a...
4	0003ad6ef0c37534f80b55b4235108024b407f0b	Fleetwood are the only team still to have a 10...	Fleetwood top of League One after 2-0 win at S...

Figure 3: Overview of Dataset

The dataset contains 3 attributes “id, article, highlights”. The first attribute id provide the unique id for each article present in the dataset. And the second attribute “article” contains long document or news article which we need to summarize. And lastly, the third attribute contains the summary of the each article in their respective highlights column, and this attribute is used as a reference while calculating the rouge score for our machine generated summary.

5. Approaches for Abstractive Text Summarization

5.1 T5 Transformer Model:

An encoder-decoder model is T5. All linguistic issues are transformed into text-to-text formats. You must first import the tokenizer and associated model using the command listed below. T5ForConditionalGeneration model should be used when both the input and output are sequences.

5.2 GPT-2 Model:

A dataset of 8 million online pages was used to train the 1.5 billion parameter language model GPT-2. The consecutive leading objective of GPT-2's training is to anticipate the successive word from the text's previous words. This straight forward aim incorporate absolute examples of plentiful jobs from distinct territory due to the diversity of the dataset. With more than 10X the parameters and trained on more than 10X the quantity of data, GPT-2 is a straight scale-up of GPT. As seen in the sample script run generation.py, this capability enables GPT-2 to produce syntactically consistent text. The previously computed key/value attention pairs can be used as input for the PyTorch models. By using this historical value, the model is prevented from recalculating previously computed values when text is generated.

5.3 Pegasus Model:

PEGASUS trains a machine to anticipate sentences by removing whole sentences from a document and the model is trained to forecast the sentences. It constructs logical to pre-train the encoder as a masked language model as PEGASUS's base building design contains of an encoder and a decoder even if the GSG is its key addition. A Modern Approach to Abstractive Text Summarization Reading a document and writing a summary is a common assignment for students.

6. Results

In the first page of application, there will be a text area where users who wants the summary of a long document will enter their text .

Figure 4: Text Entering Page

And then after clicking on summarize button, it will navigate to another page. Where the machine generated summary will be displayed.

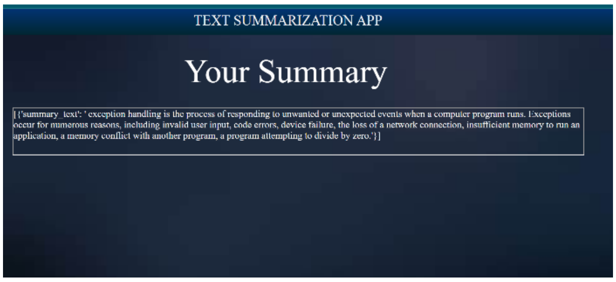

Fig:5 Summary Generated Page

6.1 Result Analysis:

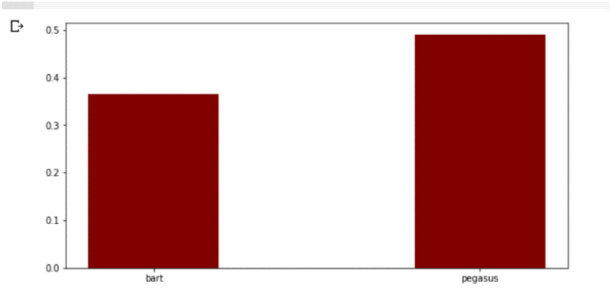

Figure:6 Comparison of Models

7. CONCLUSION

Automatic text summarization is a difficult problem, with attempts and solutions dating back a few years. as a result of the release of strong hardware and the as a result of the rise in computing power, we may use more complicated algorithms to get the desired outcome. The statistical (but computationally less expensive) extractive summarizing techniques are thus gradually giving way to abstractive summarization techniques using non-linear models (deep learning). While still a challenging undertaking, creating an abstract using the abstractive summarization method. Abstractive summarizing techniques result in more comprehensive, coherent, and information-rich summaries. The study of abstractive summarization approaches is more beneficial for the aforementioned causes. And we have evaluated the resulted summary with ROUGE [4].

To improve accuracy through the use of cutting-edge methods and machine learning algorithms. The work for using the BART model to generate a summary for a lengthy document can be enhanced and expanded.

8. REFERENCES

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

[2] Jie Hao, Xing Wang, Shuming Shi, Jinfeng Zhang, and Zhaopeng Tu. 2019. Multi-granularity self-attention for neural machine translation. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 887–897, Hong Kong, China. Association for Computational Linguistics.

[3] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In Advances in neural information processing systems, pages 1693– 1701.

[4] Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In Text Summarization Branches Out, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

[5] Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

[6] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. SummaRuNNer: A recurrent neural network based sequence model for extractive. In Proceedings of In Association for the Advancement of Artificial Intelligence.

[7] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21(140):1–67.

[8] Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Advances in Neural Information Processing Systems, volume 30, pages 5998–6008. Curran Associates, Inc

[10] Chiang, T., Che, Z. H., Huang, Y., & Tsai, C. (2022). Using an Ontology-Based Neural Network and DEA to Discover Deficiencies of Hotel Services. *International Journal on Semantic Web and Information Systems (IJSWIS), 18*(1), 1-19. http://doi.org/10.4018/IJSWIS.306748