Comparative Study on News Summarization using various Transformer Based Models

Tejas Karkera[1], Nileema Pathak[2]

[1]Student , Dept. of Information Technology, Atharva College of Engineering
[2]Professor Dept. of Information Technology , Atharva College of Engineering

---

Abstract - ***As the world is moving at a very fast pace, there is a lot of information which is amassed on a daily basis with events that happen around the world and it does happen at times when it is very time consuming to keep abreast with all this information. Hence a summarized way of interpreting things around is paramount which creates paraphrased shorter version of the long texts. News Summarization is one such methodology which tries to encompass the majority of information in news articles either by extracting the major points (Extractive summarization) [7] or by rephrasing the major points (Abstractive summarization) [8] . Building on this thought, this paper presents a comparative study of various transformer based models like T5 , BART , Pegasus on three major news dataset CNN/Daily , Multinews and XSum and contemplating their performance individually .***

***Key Words*: Summarization , NLP , Transformer , T5 , BART , PEGASUS**

1. INTRODUCTION

It can be very well contemplated that everyday there are hundreds of articles which get published either in the newspaper or online on websites. As the world around is facing numerous changes it is becoming extremely difficult for people around to be aware of almost everything which happens around them. Hence people often rely on headlines to atleast make themselves aware of the worldly affairs. Still somewhere headlines will not always give the user everything which is required from the news and hence there comes the demand to have a concise version of the news articles inkling for news summarization .News summarization as the name suggests is an approach to get the most relevant information from the news article

and still not accumulating a lot of information.Often in olden times this was considered a tedious job as the generation of summaries had to be done manually. There are a lot of drawbacks when it comes to manual writing of summaries.

Some of them are as follows :-

1. It is a very time consuming process to manually write each summary.

2. The number of summaries generated are very less in a day.

3. It is not scalable and labor utilized here can be used elsewhere.

Hence it is very important to build an efficient automated news summarizer which can handle the shortcomings of the manual news summarization. Also it is very important to ruminate on the performance of these automated models such that they are able to recognize the context precisely and are able to generate the best possible summary.

2. LITERATURE SURVEY

Language models have undergone a lot of transformation and upgradation in the past few years. With the advent of transformer based models a lot of language model based tasks have been substantiated with better performance. News summarization too has shown really good results when it comes to usage of transformer models as shown in the paper **"Automated News Summarization Using Transformers" [9]**. This paper showcases the usage of transformer models like BART, T5 and Pegasus for summarizing news articles from BBC news data and comparing them to human generated summaries. This BBC news dataset contained around 2225 news articles and performance for all the models were judged on Rouge score values which was around 0.40 for all the three models. Another work which proposed a transformer based pipeline for this task was in the paper **"News Summarization Application Based on Deep NLP Transformers for SARS-CoV-2" [10]** where the dataset used was the Covid-19 Public Media Dataset which contained news pertaining to Covid and its worldwide effects.They had done a comparative study on five models which were namely BERT, XLNet, GPT-2, BART and T5 and had contemplated their results too on Rouge score values. Their results showed that Bert, an auto-encoder based model outperformed all the other models.

2.1 Understanding the Dataset

The dataset used in this paper is a collective dataset of news articles from **CNN / Daily mail [2] , Multi-News dataset [3] and XSum [1] .** All of these datasets have a column ID , the actual written article and the human generated summary. A more detailed description for the dataset is given below :-

A. The **CNN/Daily Mail dataset [2]** is a dataset containing news articles and their highlights from CNN news channel and Daily Mail. It has around 300k unique news articles, 13k news articles for validation and 11k articles for testing. As mentioned in the Paper "*Get To The Point: Summarization with Pointer-Generator Networks*." It contains multi sentence summaries which have around 3.75 sentences or it can be 56 tokens on average.

B. The **MultiNews Dataset [3]** from the paper "Multi-News: a Large-Scale Multi-Document Summarization Dataset and Abstractive Hierarchical Model" is a dataset made from news articles extracted from newser.com. The dataset has summaries which are written by professional editors and has around 44k examples for the training set and around 5k examples each for the validation and the test set.

C. The **XSum Dataset** [1] from the paper "Don't Give Me the Details, Just the Summary! Topic-Aware Convolutional Neural Networks for Extreme Summarization" is actually a dataset containing BBC news articles from the year 2010 to 2017 and is based on a wide variety of genres like Business , Politics and Sports etc.The dataset has around 200k examples for training and around 11k examples for both validation and testing

2.2 Preparing News Dataset

1. Firstly as we are using Google colab for model training and inference purpose it would be highly unrealistic to use so many data points because colab GPU provides limited amounts of memory and can cause errors. So we would be **sampling certain amount of data from the complete dataset which will be around 10000 data points** .

2. We will create a custom make dataset function which will receive the data frame and for every row in the dataset it will use the custom tokenizer for the model and tokenize the text and finally pad the text to a certain maximum length.

3. The tokenizer would return an object which will contain the input ids assigned to the words by the tokenizer and their corresponding attention masks which would be feeded to the model.

2.3 Setting up the Model and Specifying Hyper-parameters

1. As we would be using three models BART [3] , T5 [4] and Pegasus [5] , we would be required to load their individual models and tokenizer and for this we would be using the transformer's library which will help us instantiate an object for these models.

2. We would be setting Adam as the optimizer for better learning and efficient updation of weights. Each batch of data points would be passed through the instantiated model object.

3. Some of the Hyper-parameters for the model for the Training phase would be :-

A. Learning rate which would be set to 1e-4 or 3e-4 as they were providing better results.

B. The maximum number of words to be taken for the context would be 512 or 256 according to the memory requirements and the generated summary length can be 128 or 100

4. Some of the Hyper-parameters for the model for the Validation phase would be :-

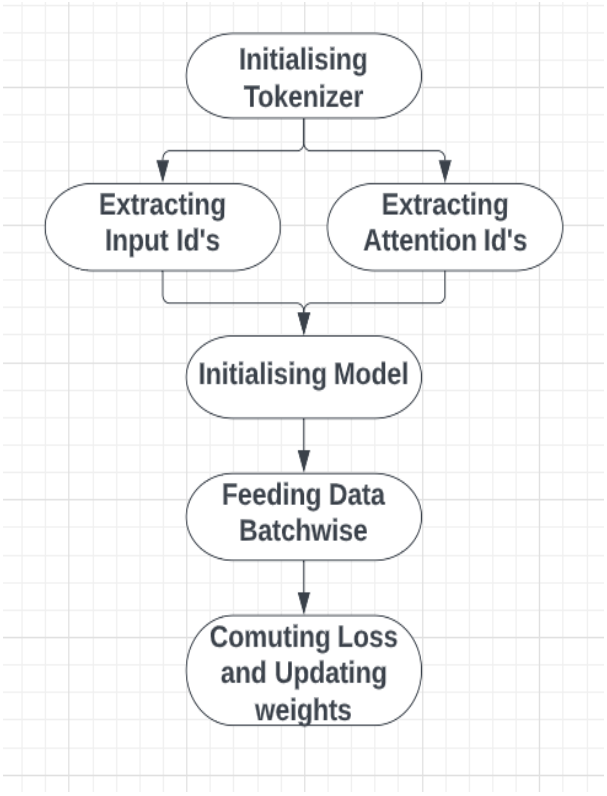
A. We have to set the repetition penalty to some value because we have to generate the best possible summary and repetition would just act as noise.The value can be set somewhere between the range 2.0 to 2.5.

B. We have to also set a penalty for the length so that if the penalty is considerably high it will produce shorter summaries.The best possible value can be somewhere around 1.0 to 1.25

2.4 Training Phase

The Dataset then would be divided into batches of sizes 4 or 6 for feeding into the model.The batch sizes have to be taken under 6 because during text summarization alot of content is being processed and hence larger batches can hinder efficient computation and can lead to out of memory error while training on Google colab.

The Source Ids and Source attention masks are then feeded into the model along with the Target Ids.After computing the loss the gradients are calculated and the weights are updates accordingly.



2.4.1 Training Flow of the Model

2.5 Validation Phase

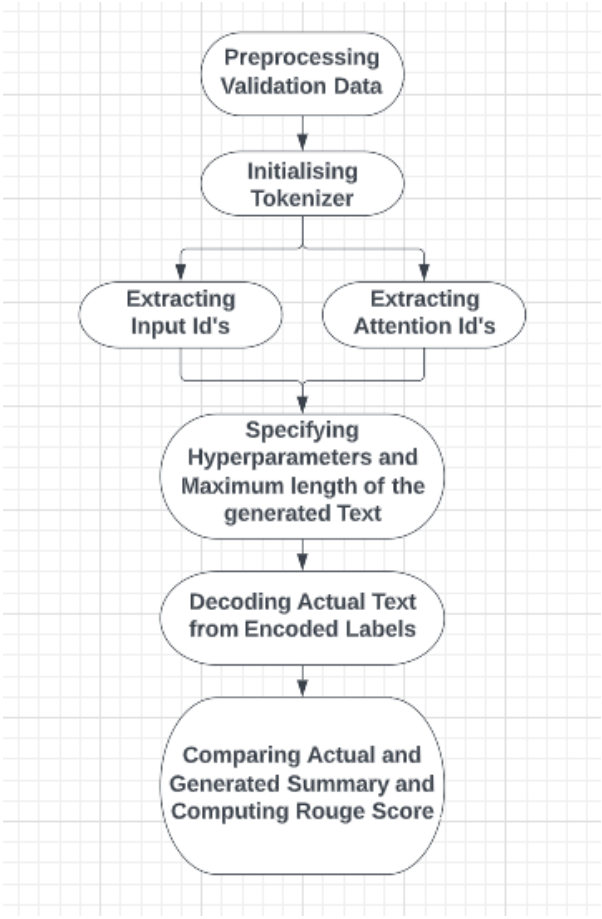
The Source Ids and Source Attention masks are extracted from the validation loader and then are used to be passed through the model to generate target Ids.The various parameters to be specified while actually generating summaries are :-

1. **maximum length** of the summary which can be in accordance with the dataset.

2. **num_beams** helps one the next n most probable words which are identified using the beam search

3. **repetition_penalty** has to also be specified because when the summaries in the dataset are smaller as in case of XSum then the model can get stuck where it is predicting the same word more then once.

4. **length_penalty** is specified to keep the length of the generated summary in a manner which is very similar to the dataset being used.



2.5.1 Validation and Inference Flow of the Model

Performance Evaluation

For performance evaluation of the generated summaries we will be using the Rouge metric [7]. Rouge in actual terms stands for Recall - Oriented Understudy for Gisting Evaluation and is actually a collection of metrics namely Rouge - 1 , Rouge - 2 , Rouge - L.

Rouge - 1 :-

It basically looks for uni-gram overlap existing between texts .

Rouge - 2 :-

This metric looks for overlaps which will be based on bi-grams

Rouge - L:-

This metric generally looks for the longest overlap which will exist between words in the text.

Table showing rouge scores for all three models.

1. CNN / DAILY MAIL DATASET

CNN / DAILY MAIL DATASET			
	ROUGE - 1	ROUGE - 2	ROUGE - L
T5	0.338	0.120	0.241
BART	0.369	0.122	0.229
PEGASUS	0.347	0.100	0.207

2. MULTI - NEWS DATASET

MULTI - NEWS DATASET			
	ROUGE - 1	ROUGE - 2	ROUGE - L
T5	0.348	0.108	0.192
BART	0.361	0.126	0.174
PEGASUS	0.364	0.111	0.187

3. **XSUM - DATASET**

XSUM DATASET			
	ROUGE - 1	ROUGE - 2	ROUGE - L
T5	0.301	0.081	0.228
BART	0.271	0.063	0.175
PEGASUS	0.280	0.072	0.201

3. Conclusion

All the transformer based models were able to produce a substantial amount of accuracy which was inferred on the Rogue Scores. Although it was very lucid that model was able to perform substantially well on almost all three datasets in terms of the Rouge - N Score. This model was able to extract and summarize the key points in the paragraphs and we were able to visualize the key attributes or words it was able to devote more attention on.

4. Future Work

Transformer models are being updated and modified every year and even newer versions of models are being developed so it is paramount to test those models too on these datasets. It is also important to understand the models performance on longer length paragraphs and then compare the generated summaries to the actual ones.This will make us understand the retainability of information of these models when the text is more.

REFERENCES

1. Narayan, Shashi, Shay B. Cohen, and Mirella Lapata. "Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization." *arXiv preprint arXiv:1808.08745* (2018).

2. See, Abigail, Peter J. Liu, and Christopher D. Manning. "Get to the point: Summarization with pointer-generator networks." *arXiv preprint arXiv:1704.04368* (2017).

3. Fabbri, Alexander R., et al. "Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model." *arXiv preprint arXiv:1906.01749* (2019).

4. Lewis M, Liu Y, Goyal N, Ghazvininejad M, Mohamed A, Levy O, Stoyanov V, Zettlemoyer L (2019) BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. CoRR abs/1910.13461:

5. Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ (2019) Exploring the Limits of Transfer Learning with a Unified Text to-Text Transformer. CoRR abs/1910.10683

6. Zhang J, Zhao Y, Saleh M, Liu PJ (2019) PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. CoRR abs/1912.08777:

7. Lin C-Y (2004) Looking for a Few Good Metrics: ROUGE and its Evaluation

8. Moratanch N, Gopalan C (2016) A survey on abstractive text summarization. pp 1–7

9. Anushka Gupta, Diksha Chugh, Rahul Katarya, et al. 2021. Automated news summarization using transformers. arXiv preprint arXiv:2108.01064

10. V.Hunar Batra , Akansha Jain , Gargi Bisht , Khushi Srivastava , Meenakshi Bharadwaj , Deepali Bajaj , Urmil Bharti . Covshorts: News Summarization Application Based On Deep Nlp Transformers For Sars-cov-2 , 2021 9Th International Conference On Reliability, Infocom Technologies And Optimization (Trends And Future Directions) (Icrito) .