Evaluating Extractive Text Summarization with BERTSUM

Stanford CS224N Custom Project

Sihguat Torres
Department of Computer Science
Stanford University
sihguat@stanford.edu

Abstract

In this paper we dive into how effective a pre trained BERT trained on a CNN and DailyMail dataset can summarize news content. The Focus is on the evaluation of the algorithm BERTSUM using metrics such as ROGUE. I prepare the CNN/DailyMail dataset, tokenize the data using Stanford CoreNLP, use the pretrained BERT model and then utilize it as stated in the paper for extractive summarization [1]. Once this is achieved I will dive deeper into the evaluation metrics through discussing ROGUE-1, ROGUE-2, ROGUE-3 and ROGUE-L. Additionally we use a naive extractive summarization for each article consisting of the first 3 sentences to use as a benchmark LEAD-3 as metrics to evaluate the summaries given by the model.

1 Introduction

The BERT Model revolutionized NLP and with its easily fine tuned parameters to different NLP tasks. In particular the task of text summarization has been researched intensively in the subfields of abstractive and extractive text summarization. The goal of extractive text summarization models is to score each sentence in the document to be able to include the most relevant sentences in the summary. In the case of abstractive summarization there is a need for the model to have word generative capabilities given words or context that might not be included in the document.

The progress in the extractive text summarization has seen remarkable accuracy thanks to models like BERTSUM which uses fine tuning layers to add document based context from the BERT outputs to more efficient models such as DistillBert which shows relatively similar performance but needs a lot less space and time to run [2].

2 Related Work

Once google released it's BERT model to the public we saw an influx of finetuning these BERT models to various NLP tasks. In particular the most extensive summary dataset with reference summaries is the CNN/ DailyMail dataset which has lead to several algorithms to use it in efforts to summarize news reports[1]. This expanded to various subsets of text summarization such as suing BERT models to extractively summarize lectures [3]. I will be using the *Text Summarization with Pretrained Encoders* paper as a guide to evaluate such a model and use their fine tuned pretrained BERT model to test the outputted summaries.

Besides extractive summarization there has been a lot of research done in a more difficult task which although more complex uses same techniques as extractive summarization: abstractive summarization. Some state of the art algorithms used in this sub field of text summarization which achieves the best results on the dataset which we will use for our experiment (CNN/DailyMail

dataset)[4].

We have even seen extractive summarization in specific subdomains of text such as medical reports. The paper [5] was able to demonstrate great results by fine tuning this specific type of reports.I hope to get similar accuracy with the predictions and expand on other evaluation methods and what it tells us about extractive summarization in news article summarization.

3 Approach

My contribution in this project is to expand on the evaluation done on the pre-trained BERT model. I make sure to use the summarization layers on the BERT output embeddings as outlined in the paper [1]. As done in the paper I will evaluate the predicted extractive summaries of the model by using Automatic Summarization or the ROUGE set of metrics. The evaluation metrics used to show how accurate the model performs against the scores that were given in the paper and use the LEAD-3 metric, described in 4.2, as a benchmark. After doing so I will show differences in evaluation scores as described by the BERTSUM model in [1], in section 4.3, then expand on what these metrics tell us about the model's perception in NLP at large in section 3.

3.1 BERTSUM Architecture

The BERTSUM model which is an extension of the BERT model but in particular to the task of text summarization. The main difference between BERT and BERTSUM is the addition of inputting data with symbols to represent start and end of sentence so that the model may learn sentence representations. Additionally another difference is in the segment embeddings and how BERTSUM embeds pairs of sentences to learn adjacency patterns between each input sentence. The overall goal of this model is to give every sentence in the document a score representing the relevance of the sentence to the overall document, as a way to indicate to the model which sentence should be included in the summary.

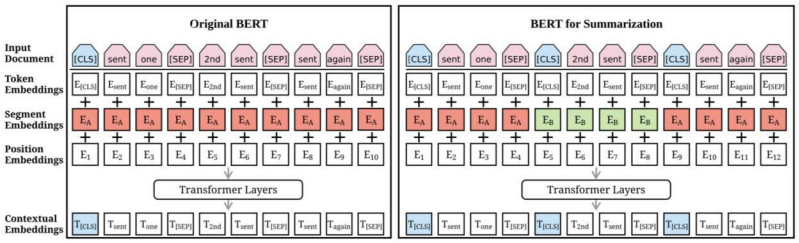

Figure 1: BERT Architecture: BERTSUM vs BERT

As previously mentioned BERTSUM builds on the outputs of BERT and thus is finetuned for the task of text summarizing so there is a need for these fine tuning layers. The fine tuning take advantage of the objective of summarization to place positional embeddings indicating location of the sentence in the document.

$$\tilde{h}^l = \mathrm{LN}(h^{l-1} + \mathrm{MHAtt}(h^{l-1}))$$
$$h^l = \mathrm{LN}(\tilde{h}^l + \mathrm{FFN}(\tilde{h}^l))$$

Figure 2: BERT Architecture: Fine Tuned L-layer Transformer

Figure 2 shows the Transformer layer and l is the lth layer of the transformer. Additionallt $h_0 = PosEmb(T)$ where T is the output of the BERTSUM model and the positition embeddings are a way to tokenize the output in a way that shows document contextualization. After passing it through L layers of this transformer we apply a score with h_i^L. This equation demonstrates how we utilize the classifier to obtain the score $\hat{y}_i$. By ranking these scores we are able to conclude on the most relevant (highest $\hat{y}_i$) sentences which then will make up the summary.

$$\hat{y}_i = \sigma(W_o h_i^L + b_o)$$

Figure 3: BERT Architecture:Scoring

4 Experiments

As mentioned as used in the paper [1] the benchmark that was used is the LEAD3 metric. This will lead to preprocessing of articles to get these naive summaries (first three sentences) and then also get all summaries in a file so that we then can utilize a rouge evaluation to see the performance on the BERT model in comparison to the BERSUM model depicted in [1].

4.1 Data
The BERTSUM architecture that was discussed in section 3 will be utilized to give us the output predicted summaries. These predicted summaries will be evaluated against the golden summaries as reference, which can be found in the CNN/Dailymail testing data. The summerizer is already pre-trained and the outputs are directly used by training on a BERTSUM model. The CNN/Dailymail dataset is first processed by tokenizing it to feed it into the BERTSUM [1]. This consists of including multiple [CLS] to accommodate sentence pattern recognition as well.
with h_i^L. The data can be downloaded through github [4], used StanfordCoreNLP to break up into

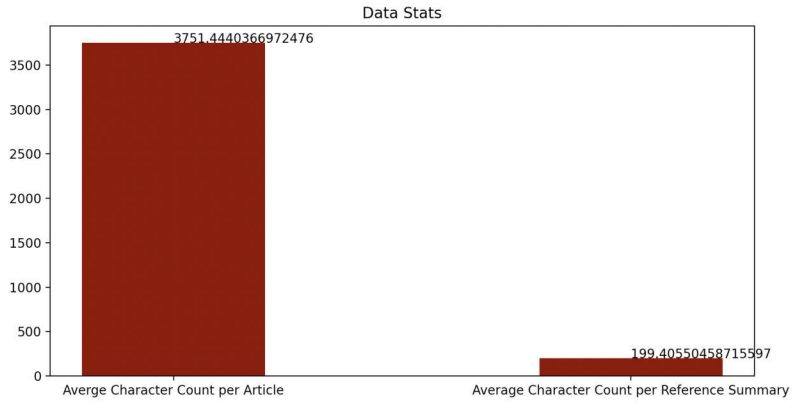

Figure 4: Data Stats: Size of Articles versus Size of Summary

sentences, and the preprocessed using the tokenization used in, then finally using the BERT text summarizing model utilized in this paper [4].

4.2 Evaluation Method To evaluate our predicted summaries we will choose the top 3 highest scored sentences and use them as our summary. As seen in the paper the LEAD-3 metric will be used to display a naive prediction of the summary and I will use the NLTK documentation [7] to parse the document in hopes to get the first 3 sentences that we will use in our summary.

For evaluation I used a series of ROGUE metrics where I show the f1-score, the precision and the recall. The three metrics that I decided to show since they were used by the paper [1] where the the unigram, bigram and longest common subsequence evaluation methods (ROGUE-1, ROGUE-2, ROGUE-L) as a form to analyze how well our model performed with the CNN and DailyMail test data. These figures were calculated using the rouge library [8] and made sure to put predicted summaries in a separate file to be able to use the textfile average of multiple summaries and reference summaries for the 1090 test articles in the CNN dataset.

4.3 Results
Our results test the precision and recall for different metrics designed around predicted summaries. First we show the size of summaries between LEAD-3 and BERTSUM. We see that on average the LEAD3 summaries are twice as large as the reference summary.

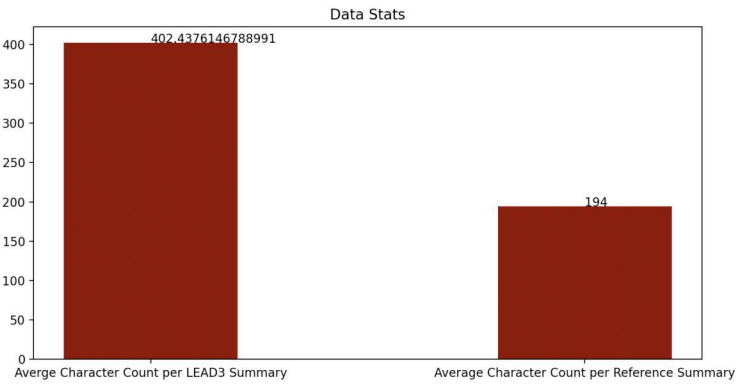

Figure 5: Data Stats: Size of Reference Summary versus predicted Summary through LEAD3 Summary

Out of the three scores that are seen in our results, precision and recall are the most important. Recall is the sensitivity or the ratio of correctly predicted summary n-grams to all the n-grams in the summary. While on the converse precision is the ratio of correctly predicted n-grams to the total predicted n-gram. The f1 score is a weighted combination of both metrics so hard to quantify but we see that the precision and recall are increased by the BERTSUM model. Below shows these results for each metric: ROUGE-1, ROUGE-2 and ROUGE-L. All of the rouge scores are averages over the rouge score between the reference summary against the BERTSUM predicted summaries (As depicted as BERTSUM Summary in table) and reference summaries against LEAD3 summaries.

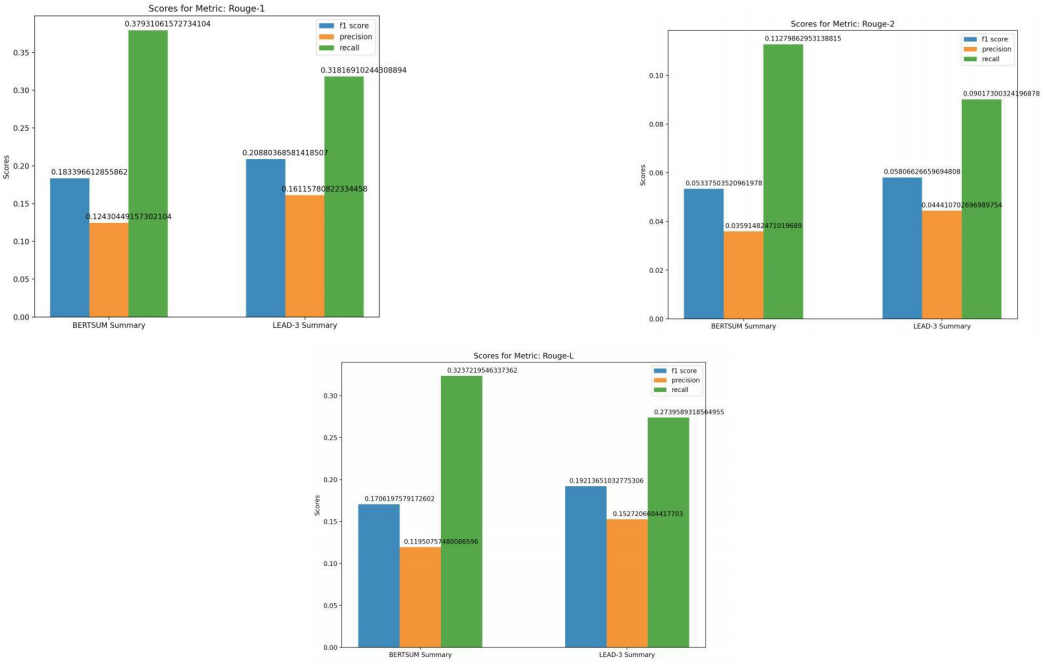

Figure 6: Data Stats: Size of Reference Summary versus predicted Summary through LEAD3 Summary

We see that for the recall measure the BERTSUM model is able to outperform our benchmark of LEAD3. Additionally we see that it gives the highest scores on Rouge-1 over the other metrics Rouge-2 and Rouge-L. A pretty large improvement from LEAD3 considering that the dataset are news articles which tend to get right to the point and thus having much more context at the beginning of the artcle.

5 Analysis

Some of the error in a not as effective results by the BERTSUM model is that the predicted summaries would majority of the time include the first sentence but would shift away from LEAD3 afterwards which begs to the question if this this is due to the type of documents that we are analyzing or because of a mistake in training. I believe that it's due to the sort of data that we analyze and the introduction to such articles being a good summary or intro into what will be talked about in the remaining article. Additionally precision figures are down for this same reason that it constantly picks a similar type of prediction by paying much more attention to the initial sentences.

6 Conclusion

It was easily seen before our results that the .05 (Figure 4) ratio from document length to summary tells us the impact of a high precision and recall model can offer. The ability to understand a piece of text in a much a fraction of .05 of the original time average shows how important NLP extractive summarization in multiple fields.

Another point is that evaluating these text summarizing models through ROUGUE metrics or BLEU scoring for question answering are hard to quantify, meaning difficult to see how effective these models are and what figure is "good enough."

Overall the precision of the identified BERTSUM was lower than expected while Recall has a significant increase than our benchmark, as expected. The precision figures went against my original hypothesis that BERTSUM would outperform LEAD3 but as referred in section 5 the type of data that we are testing and the consistency of picking the first sentence leads to a drop in precision.

7 Future work

As initially planned this project was going to be designed around financial reports but as I researched these sort of datasets with golden summaries were extremely scarce thus building further databases would advance the complexity of these models. Thus an important factor in training these models on specific domains of text is to create specific datasets with reference summaries. In the example with medical text summarization [4] these BERT techniques were used for a very specific domain of texts so it's crucial that datasets of financial reports and their summaries are created to expand on the summarization efforts in finance along with any other specific field. Additionally, future work that would largely benefit the text summarization field would be further work in advancing abstractive summarization. This is because the use of abstractive summarization includes outside vocabulary and ideas through longer scripts and can include more valuable summarizations than extractive summarization.

References

1. Liu, Y., Lapata, M. (2019). Text summarization with pretrained encoders. arXiv preprint arXiv:1908.08345.
2. Sanh, V., Debut, L., Chaumond, J., Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108.
3. Miller, D. (2019). Leveraging BERT for extractive text summarization on lectures. arXiv preprint arXiv:1906.04165.
4. Vinod, P., Safar, S., Mathew, D., Venugopal, P., Joly, L. M., George, J. (2020, June). Fine-tuning the BERTSUMEXT model for Clinical Report Summarization. In 2020 International Conference for Emerging Technology (INCET) (pp. 1-7). IEEE.
5. Dou, Z. Y., Liu, P., Hayashi, H., Jiang, Z., Neubig, G. (2020). GSum: A general framework for guided neural abstractive summarization. arXiv preprint arXiv:2010.08014.
6. https://github.com/JafferWilson/Process-Data-of-CNN-DailyMail

7. http://www.nltk.org/
8. https://pypi.org/project/rouge/