Comparison of Abstractive Summarization Input Reduction Methods

Nicholas Schantz
UC Berkeley
nschantz@berkeley.edu

Abstract

Abstractive summarization tasks using transformer-based architectures are limited by their poor ability to scale with input size. This paper examines reducing document size as a preprocessing step to this task as a possible solution to this limitation. We compare 5 reduction methods isolating effects of locality of information, input sentence ordering, and magnitude of reduction. We find reducing document size by 50% while maintaining ordering and leveraging information locality results in model output comparable to utilizing the unaltered input document. These findings imply input reduction may be a cost-effective approach to preliminary research and productionization of summarization models.

1 Introduction and Background

In the task of abstractive summarization, transformers represent the state of the art architecture. However, some of the most popular implementations of these transformer-based models limit their input dimension space to 512-1024 input tokens. This is born out of a practical consideration, as the attention mechanism core to transformers scales quadratically with sequence length [10].

This creates a limitation on the types of documents that can be summarized, and is exacerbated by the tokenization function, which can turn a single word into several tokens. The standard solution is to truncate these tokens to the input size of the model, although an increasingly popular solution is to fine-tune existing models that allow for longer inputs by altering the attention mechanism to scale more efficiently [10].

Relatively little research exists assessing the performance of abstractive summarization models when given reduced document inputs.

Xinwei Du, et al (2021) take the approach of creating extractive summaries as a pre-processing step using the TextRank algorithm as an input to large input abstractive summarization models, BigBird, and LED. Additionally, they perform post-processing on model output using GPT-3 [11].

We seek to understand the relationship between document reduction preprocessing methods and the subsequent change in model performance using simpler extractive methods.

More formally, we seek to answer the questions: *Does a reduction in input materially affect the quality of abstractive summarization? What is the sensitivity of summarization quality metrics to input reduction methods and magnitude?*

2 Methods

The tasks at hand to answer these questions are first, to reduce the document size, and second, to evaluate the change in

ROUGE and Coherence metrics in relation to those document reduction methods.

2.1 Data

We use the CNN/DailyMail dataset for our assessments [12]. We chose this dataset mainly based on its popularity in summarization tasks. Additionally, this dataset has the attractive characteristic of containing articles long enough to potentially benefit from reduction methods.

However, these methods could extend to any dataset and may be more useful for datasets of much longer texts.

2.2 Evaluation Metric and Baseline

For our main metrics of evaluation of the summaries, we use ROUGE, document-level Coherence, and inter-document cosine similarity between model outputs.

Rouge: Rouge1, Rouge2, and RougeL [2] are adopted as the standard metrics for abstractive summarization tasks.

Coherence: A secondary metric used to measure the coherence of the output summaries is defined below. Outlined by Nayeem and Chali (2017), this metric uses the weighted sum of Named Entity overlap and cosine similarity between adjacent sentences, which is then averaged over the document.

$$Coherence = \frac{1}{n-1}\sum_{i=1}^{n-1} Sim(s_i, s_{i+1}),$$

$$Sim(s_i, s_{i+1}) = \lambda \times NESim(s_i, s_{i+1}) + (1 - \lambda) \times CosSim(s_i, s_{i+1})$$

$$NESim(s_i, s_{i+1}) = \frac{\left|NE(s_i) \cap NE(s_{i+1})\right|}{min\left(\left|NE(s_i)\right|, \left|NE(s_{i+1})\right|\right)}$$

$$CosSim(s_i, s_{i+1}) = \frac{s_i \bullet s_{i+1}}{\|s_i\| \|s_{i+1}\|}$$

Inter-Document Cosine Similarity: We take the document-level cosine similarity between summaries as a tertiary metric to evaluate diversity of outputs and closeness to the human-generated summary.

Baselines

We have included two baselines: the model output using the unaltered text as an input to the model (referred to as Baseline) and the Lead3 model, a popular extractive baseline for this dataset [13].

2.3 Model

The model used for generating all summaries is the pre-trained abstractive summarization model for DistilBART, available through HuggingFace [8].

This model is based on a distilled checkpoint of the BART model [2][3]. We chose this model for its relatively small size and inference speed, and as a practical consideration given the large number of summaries generated. We used Google Colab Pro with GPU acceleration for all coding tasks. Model details are in the Appendix and accompanying code.

Because this model is pre-trained on the CNN/DailyMail dataset, we chose to use the test split to better understand the out-of-sample performance of these methods.

Please note, for the coherence and cosine similarity metrics we use a random subsample of outputs as a practical consideration, given the computational time and cost of calculating these metrics.

2.4 Strategies

We took three main approaches to text reduction. In each case, we started by splitting the unstructured text into sentences. We then extracted sentences

from the original text to create 50% reduced texts as follows:

- EveryOther: Extract every other sentence
- BegMidEnd50: Extract 50% of the sentences, equally sampled from the beginning, middle, and end of the array of sentences, maintaining the original order
- Random50*:* Extract 50% of the sentences randomly, without maintaining the original order

These extraction methods help isolate some structural characteristics of the input documents, which we will discuss later.

We then created two additional reduced datasets, following the same method of the BegMidEnd50 dataset.

- BegMidEnd25: Extract 25% of the sentences
- BegMidEnd75: Extract 75% of the sentences

This allows us to directly assess the effects of the magnitude of text reduction on summary quality.

3 Results and Discussion

The following data was collected for each method of text reduction and the baselines:

- ROUGE1, ROUGE2, and ROUGEL (f measure, precision, and recall) for each observation in the dataset, using the human-generated summary for comparison
- Coherence score weighting the Named Entity term as 0 and .5 for each output summary
- Cosine similarity between all pairs of summary outputs as well as the human-generated summary

For these data collections we report the overall results and provide additional results in the Appendix.

Selection Methods

First, we compare the methods resulting in a 50% reduction in the input document: EveryOther, BegMidEnd50, and Random50.

We compare these methods using ROUGE metrics against the human-generated summaries. In Figures 1 and 2, we show the average f measure of each method. Charts share color-encoding by reduction methods.

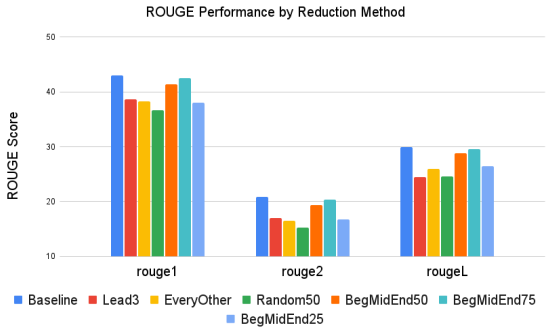

Figure 1: Comparison of ROUGE scores

The Lead3 baseline's outperformance over methods Random50 and EveryOther suggests the majority of the information relevant to a summary is in the first three sentences of the article. This also suggests there is relatively limited marginal information evenly spread throughout the rest of the document.

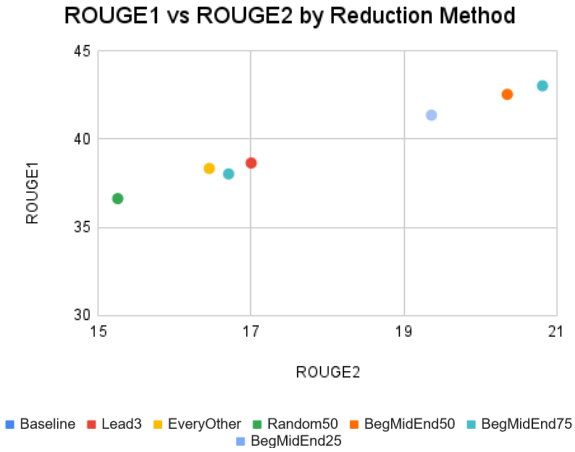

Figure 2: Comparison of ROUGE1 and 2 by method

The outperformance of EveryOther to Random50 suggests maintaining sentence

order does matter to summarization in reduced inputs. The outperformance of BegMidEnd50 to other methods suggests that information relevant to summarization is clustered in the document.

Figure 3 shows the average precision and recall of reduction methods against the baseline models. Notably, the BigMidEnd50 reduction method has as good of, or better, precision than either baseline summary. However, the BigMidEnd50 output recall is significantly worse than the baselines across all ROUGE n-gram sizes. This suggests that providing abstract summarization models with reduced inputs may result in summaries with more relevant content; however, these summaries may also suffer from excessive dreaming, relative to the baseline models.

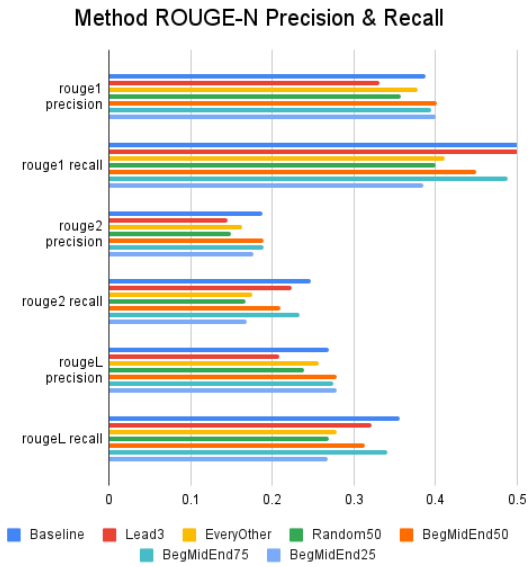

Figure 3: ROUGE component by Method

Next, we examine the coherence scores of these methods and the order of those coherence scores, in figure 4. Coherence scores when weighting Named Entity terms as 0 and .5 are included to assess the impact of Named Entities on coherence labeled as Coherence0 and Coherence50, respectively.

There is little variance in coherence scores across methods, including the human-generated summary. Notably, the Lead3 coherence score is much greater than any other summary in both Named Entity weightings, indicating any abstractive summarization leads to a drop in coherence relative to the input document. The Random50 summary coherence score is also notably close to the baseline, indicating the summarization model's resilience to input sentence order and reduced input.

A sample of the different summaries is provided in the Appendix in Table 1. Although this is anecdotal evidence, by our qualitative assessment of the summaries, all machine-generated summaries are more informative of the original article than the Lead3. Furthermore, BegMidEnd50 appears to provide a summary comparable to the Baseline.

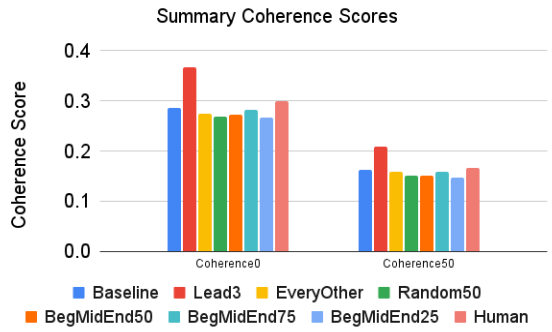

Figure 4: Comparison of coherence scores

We next examine the cosine similarity between the human-generated summary and each machine-generated summary. Results are shown in Figure 5.

The outperformance of the Baseline, BegMidEnd, and Lead3 summaries relative to EveryOther and Random50 summaries again suggests that information locality is a major factor in replicating human-generated summary quality. The small variance in these metrics is further evidence of the

underlying language model's resilience to reduced inputs.

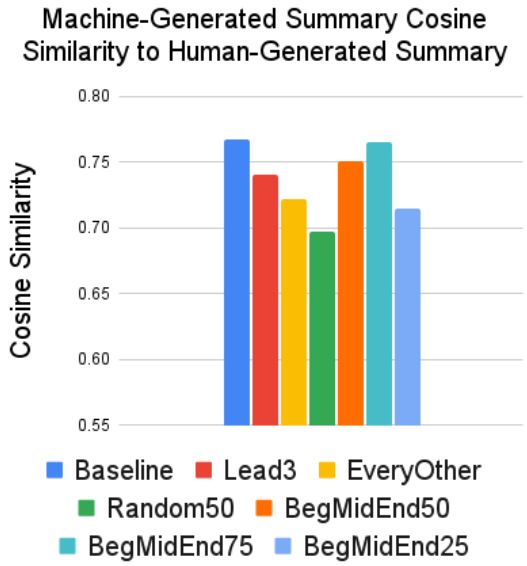

Figure 5: Comparison of cosine similarity

Further Reduction

Extending the reduction method of BegMidEnd50, we reduce the input text down to 75% and 25% of the original document as BegMidEnd75 and BegMidEnd25, respectively, to isolate the effect of input size reduction while leveraging the importance of locality in information relevant for summarization discovered using BegMidEnd50.

Unsurprisingly, ROUGE metrics, shown in Figure 2, degrade with further input document reduction. What is notable is the non-linear nature of the decay. One possible explanation for this is the inequality of information present in the beginning, middle, and end sections of the original document.

This idea is supported by the similar ROUGE performance of the Lead3 summary, which represents about 10% of the original text, to the BegMidEnd25 output. Although there is relevant information clustered in the middle and end sections of a document, the beginning is disproportionately relevant for summarization.

Looking at figure 4, we can see that the decline in BigMidEnd25's ROUGE score can largely be attributed to a decline in recall. We find it intriguing that the precision of BigMidEnd25 is greater than or equal to that of either baseline. This would suggest that reducing input to 25% of its original size allows the model to focus on the most relevant information. Understandably, this also likely leads to the greatest amount of dreaming by the model to fill in the gaps left by the reduction.

As shown in Figures 4 and 5, coherence and cosine similarity between the human generated summary and machine-generated summaries do not drop off significantly with further reduction. This, again, demonstrates the model's ability to produce viable summaries given limited inputs.

In our opinion, based on the sample summary in Table 1, in the Appendix, summaries generated using BegMidEnd75 and BegMidEnd25 produce better summaries than the Baseline model and even the human-generated summary. Although this is anecdotal evidence, this is a very interesting finding, as pre-processing appears to allow the summarization model to focus on the most important elements of the original document, potentially resulting in higher quality outputs.

Final results

- Reducing summarization model inputs by 50% using the BegMidEnd method produces summaries comparable to those produced when using the unaltered document as measured by ROUGE and Coherence scores

- Maintaining document sentence order in model input proves to be important to summarization quality
- There is evidence to suggest a strong locality element to information relevant to abstract summarization

Limitations and Future Work

This work has focused on news articles using a single model trained on news articles, which have more predictable structure than some other types of documents. Therefore , it is unclear whether or not these results will be robust to diverse writing styles, input document length, and model architectures.

Future work should center around testing robustness of the BegMidEnd reduction methods, and the performance of summarization models fine-tuned on reduced inputs.

Should the relationship between input reduction and summarization quality prove robust, reduced input summarization could be used as a computationally- and cost-effective baseline in model development.

Although less certain, and requiring human assessment, summaries generated using the BegMidEnd25 input may provide more succinct and informative outputs. If these results are consistent, significant localization-based input reduction may be a viable method to improve summary output quality.

4 Conclusion

Similarity in ROUGE metrics show reducing input texts by 50%, while maintaining order and selecting structurally important sections, does produce a summary of lower yet similar quality.

As a practical consideration when productionizing transformer-based abstractive summarization models, employing reduction methods as a pre-processing step for inference inputs may improve overall system performance with relatively little degradation in output quality.

References

1. 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.
2. 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.
3. Shleifer, S., & Rush, A.. (2020). Pre-trained Summarization Distillation.
4. Shashi Narayan, Shay B. Cohen, & Mirella Lapata (2018). Don't Give Me the Details, Just the Summary! Topic-Aware Convolutional Neural Networks for Extreme Summarization. *ArXiv, abs/1808.08745.*
5. See, A., Liu, P., & Manning, C. (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)* (pp. 1073–1083). Association for Computational Linguistics.
6. Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, & Phil Blunsom (2015). Teaching Machines to

Read and Comprehend. In *NIPS* (pp. 1693-1701).
7. Mir Tafseer Nayeem and Yllias Chali. 2017. Extract with Order for Coherent Multi-Document Summarization. In *Proceedings of TextGraphs-11: the Workshop on Graph-based Methods for Natural Language Processing*, pages 51–56, Vancouver, Canada. Association for Computational Linguistics.
8. *SSHLEIFER/Distilbart-CNN-12-6 · hugging face*. sshleifer/distilbart-cnn-12-6 · Hugging Face. (n.d.). Retrieved July 30, 2022, from https://huggingface.co/sshleifer/distilbart-cnn-12-6
9. *NSCHANTZ21/cnn_dailymail-parsed-processed · datasets at hugging face*. nschantz21/cnn_dailymail-parsed-processed · Datasets at Hugging Face. (n.d.). Retrieved July 30, 2022, from https://huggingface.co/datasets/nschantz21/cnn_dailymail-parsed-processed
10.
11. Beltagy, I., Peters, M., & Cohan, A.. (2020). Longformer: The Long-Document Transformer.
12. Du, X., Dong, K., Zhang, Y., Li, Y., & Tsay, R.Y.. (2021). The Influence of Data Pre-processing and Post-processing on Long Document Summarization.
13. Nallapati, R., Zhou, B., Santos, C., Gulcehre, C., & Xiang, B. (2016). Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond.
14. Dohare, S., Karnick, H., & Gupta, V.. (2017). Text Summarization using Abstract Meaning Representation.

Appendix

Model	Summary
Baseline	Jordan Spieth secured his first ever major title at Augusta National on Sunday . The 21-year-old won the 79th Masters at Augusta on Sunday afternoon . The US Open champion was crowned Masters champion at the end of the tournament . Spieth has been criticised for his lack of experience in the game .
Lead3	Sporting America fell in love with him because of his skill under pressure and his sheer bravura. The rest of America fell in love because of what happened after the winning putt was holed. The 79th Masters attracted massive ratings in the USA and there is no surprise in that.
EveryOther	Jordan Spieth secured his first major title at Augusta National on Sunday . The 21-year-old won the 79th Masters on 14-under-par after finishing tied for second . Spieth has killed forever the idea that you have to have a cold heart and a ruthless edge to be a winner . Along with Rory McIlroy, Spieth is going to carry it for the next decade and what an exciting time it promises to be .
Random50	The rest of America fell in love because of what happened after the winning putt was holed . Spieth just felt he was ready and it was not long before he was showing why . Sporting America fell into love with him because of his skill under pressure and his sheer bravura .
BegMidEnd50	Jordan Spieth secured his first ever major title at Augusta National on Sunday . The 21-year-old won the 79th Masters with a holed putt to holed the 13-under par on the 18th green . Spieth embraced his family after the win and gave an unscripted speech to the crowd . Politicians who try so hard to be popular could learn much from Spieth . The game is very fortunate to have a man like McIlroy, who is four years older and less than a pro for less than two years .
BegMidEnd75	Jordan Spieth secured his first ever major title at Augusta National on Sunday . The 21-year-old won the 79th Masters with a birdie putt on the third green on the final day . Spieth has banked more than $13million (£8.86m) on the PGA Tour in 18 months . He is No 2 behind Rory McIlroy in the world after winning the US Open .
BegMidEnd25	Jordan Spieth won the Masters at Augusta National on Sunday . The 21-year-old is now No 2 in the world behind Rory McIlroy . Spieth has banked more than $13million (£8.86m) on the PGA Tour in 18 months .
Human	Jordan Spieth won his maiden major at the age of 21 at Augusta . The 79th Masters attracted massive television ratings in the USA . Spieth went back onto the 18th green to formally acknowledge the crowd . Jack Nicklaus and Phil Mickelson are among those to pay tribute to Spieth .

Table 1: Summary Sample

Model	rouge1	rouge2	rougeL	CosSim to Human	Coherence0	Coherence50
Baseline	43.0	20.8	29.9	0.77	0.29	0.16
Lead3	38.6	17.0	24.4	0.74	0.37	0.21
EveryOther	38.3	16.5	26.0	0.72	0.27	0.16
Random50	36.6	15.3	24.6	0.70	0.27	0.15
BegMidEnd75	42.5	20.4	29.6	0.77	0.27	0.15
BegMidEnd50	41.4	19.4	28.8	0.75	0.28	0.16
BegMidEnd25	38.0	16.7	26.5	0.71	0.27	0.15
Human					0.30	0.17

Table 2: All summary metrics. ROUGE metrics are f measures. CosSim to Human is the Cosine Similarity of the summarization to the Human-generated summary. Coherence0 is the coherence score of the summary output, weighting the Named Entity component of the coherence score as 0. Coherence50 is the coherence score of the summary output, weighting the Named Entity component of the coherence score as .5

ROUGE-N Score Components								
rougeN	**Metric**	**Baseline**	**Lead3**	**EveryOther**	**Random50**	**BME50**	**BME75**	**BME25**
rouge1	**precision**	0.388	0.331	0.378	0.358	0.402	0.394	**0.401**
	recall	**0.511**	0.505	0.411	0.400	0.450	0.489	0.385
	fmeasure	**0.430**	0.386	0.383	0.366	0.414	0.425	0.380
rouge2	**precision**	0.188	0.145	0.163	0.149	0.189	**0.189**	0.177
	recall	**0.247**	0.224	0.176	0.167	0.210	0.233	0.169
	fmeasure	**0.208**	0.170	0.165	0.153	0.194	0.204	0.167
rougeL	**precision**	0.269	0.209	0.256	0.240	**0.280**	0.274	0.279
	recall	**0.356**	0.322	0.279	0.270	0.314	0.341	0.268
	fmeasure	**0.299**	0.244	0.260	0.246	0.288	0.296	0.265

Table 3: Average Rouge Metric Components by Reduction Method. Highest values by row are bolded