Exploring the limits of a base BART for multi-document summarization in the medical domain

Ishmael Obonyo
Universitat Pompeu Fabra
University of Nairobi
ishmaelny@gmail.com

Silvia Casola
Università degli Studi di Padova
Fondazione Bruno Kessler
scasola@fbk.eu

Horacio Saggion
Universitat Pompeu Fabra
horacio.saggion@upf.edu

Abstract

This paper is a description of our participation in the Multi-document Summarization for Literature Review (MSLR) Shared Task, in which we explore summarization models to create an automatic review of scientific results. Rather than maximizing the metrics using expensive computational models, we placed ourselves in a situation of scarce computational resources and investigate the limits of a base sequence to sequence models (thus with a limited input length) to the task. Although we explore methods to feed the abstractive model with salient sentences only (using a first extractive step), we find that the results still need some improvements.

1 Introduction

To summarize medical knowledge on specific issues, researchers undertake systematic reviews of the available literature. The process is usually long and expensive; it requires identifying appropriate studies, critically interpreting their findings, and finally synthesizing the results.

Recently, Natural Language Processing (NLP) researchers have explored the use of automatic text summarization models and tools to assist researchers with the process. Previous works by DeYoung et al. (2021); Wallace et al. (2021) have tried to model the problem as a multi-document summarization task, where several input papers (or abstracts) are summarized to generate review conclusions. Summarizing several documents is challenging, and few resources exist (DeYoung et al., 2021) compared to single-document summarization tasks.

The shared task of Multi-document Summarization for Literature Review (MSLR) adopted a similar approach and challenged participants to explore the state-of-the-art systems with two large-scale multi-document summarization datasets for literature review. To this end, instead of aiming at using very complex models to maximize the target metrics, we place ourselves in a situation of scarce computational resources and explore the limits of a base sequence-to-sequence model, BART, to the task. Our contributions to this shared task, therefore, are as follows:

- We explore the performance of a simple base transformer, namely BART, for this task.
- We explore ways to deal with the limited input size of such models, applying an extractive step before the abstractive one.
- We aim at creating general models, and explore how the two datasets can be combined during training to improve performance.

After analyzing the datasets (Section 2), we first experiment with baseline models (Section 3.1); since the model can only deal with a limited number of input tokens, we explore various strategies to reduce the input size (Section 3.2).

2 Datasets and metrics

We evaluated the models on two datasets:

Cochrane (Wallace et al., 2021): The dataset consists of 4,692 systematic reviews from the Cochrane collaboration[1]. The target is the "authors' conclusions" of the systematic review abstracts, while the input is a set of titles and abstracts of the related clinical trials.

MS^2 (DeYoung et al., 2021): is built from papers in the Semantic Scholar literature corpus (Ammar et al., 2018). It consists of 17,876 reviews. The dataset also contains some background text derived from the reviews. The dataset creation was semi-automatic: for each review, each sentence is classified as background, target or other and sentences are then aggregated.

[1] https://www.cochrane. org/

Proceedings of the Third Workshop on Scholarly Document Processing, pages 193–198
October 12–17, 2022.

Table 1 reports some statistics of the two datasets. Notice that the Cochrane dataset contains some input documents for which no abstract is provided.

Results are evaluated using:

ROUGE (Lin, 2004) (ROUGE-1, ROUGE-2, ROUGE-L): These are classical metrics for summarization, and compute the token overlap between the prediction and the gold-standard in terms of n-grams and longest common subsequence. The higher the value the better the score.

BERTScore (Zhang* et al., 2020): Instead of computing exact matches, this metric considers contextual embeddings (as generated by BERT (Devlin et al., 2019)); after computing the cosine similarity among each pair in the generated sequence and the gold standard, the maximum similarities over the gold-standard tokens (Recall) and the generated tokens (Precision) are summed and normalized; they are later used to compute f1-like metric. The higher the value the better the score.

Δ EI (DeYoung et al., 2021): It is a model-based metric; the disagreement of (Is, Os, EI) triplets between the input studies and the generated summary is considered, where Is are the Interventions, Os are the Outcomes and EI is Evidence Inference. The measure aims to better correlate with the factuality of the generated summary with respect to the sources. The lower the value the better the score.

3 Experiments and results

In this work, we explore the use of a simple BART base model (Lewis et al., 2020) – that we leave unchanged – for the task of multi-document summarization.

The BART model is limited to input size of 1024 sub-token. However, as figure 1 shows above, concatenating the abstracts leads to very long input sentences, that cannot be dealt with by the model. To this end, we explore if performing a previous extractive step improves performance. Since the target text summarizes the findings of previous work, we also explore the use of a classifier to extract results only from the input.

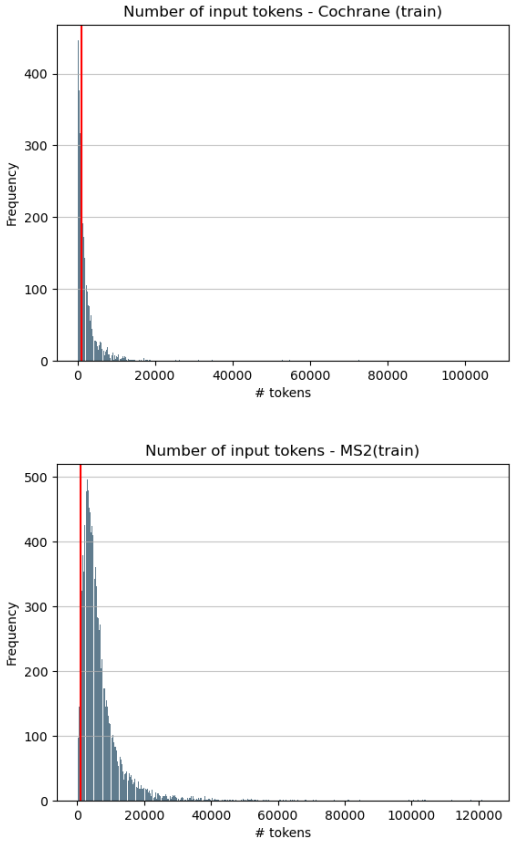

Figure 1: The number of token in the Cochraine and in the MS^2 datasets with concatenated inputs

3.1 Baselines

We train a base BART model, fine-tuned for 4 epochs on the Pubmed summarization dataset[2] (Cohan et al., 2018) to predict the target given the concatenated abstracts. Specifically, we use the concatenated abstracts as input and the target as output. We do not generally use the titles, with a few exceptions in case no abstract is present. For MS^2, we do not use any additional background information, as we want to construct models that are as general as possible. We separate the inputs using the <sep> special tokens. We do not perform any other preprocessing to the dataset text. Table 2 reports the results for our base configuration on the validation set. We report results for all metrics.

3.2 Unsupervised algorithms for decreasing the input size

Since the base model can only process a fraction of our very long input, we explore if performing an extractive step can improve performance, fol-

[2]Model *mse30/bart-base-finetuned-pubmed* from the Hugging Face model hub

	C train	C dev	C test	M train	M dev	M test
Number of input docs	40,497	5,033	5,678	323,608	5,033	5,678
Number of empty abstracts	2,611	464	470	0	0	0
Number of targets	3,752	470	470	14,188	2,021	1667
Number of docs per target (avg)	10.79	10.71	12.08	22.81	24.24	25.63
Number of tokens per abstract (avg)	224.33	222.47	14.88	299.88	302.83	301.42
Number of tokens per target (avg)	67.78	69.9	-	61.28	61.05	-

Table 1: Statistics on the Cochrane (C) and the MS^2 (M) datasets

Trained on	Eval on	R-1	R-2	R-L	BertScore	ΔEI avg	ΔEI macro
M	M	13.18	1.31	10.17	83.2	50.22	42.53
C + M (mix)	M	13.18	1.31	10.18	83.2	50.22	42.53
C, M (sequential)	M	13.23	1.35	10.18	83.14	49.33	42.55
C	C	22.48	6	16.43	86.81	31.03	38.23
C + M (mix)	C	22.86	6.03	16.82	85.1	27.85	36.44
M, C (sequential)	C	18.78	2.77	12.97	84.52	36.54	37.22

Table 2: Baseline results obtained with a base BART model on the raw input. Some models are trained on the MS^2 dataset (M) or on the Cochrane dataset (C) independently. We also trained a single model with the mixed MS^2 and Cochrane data, in random order (mix) and evaluate it on both datasets independently. Finally, we experiment with sequential fine-tunings over the two datasets (with the fine-tuning over the target dataset being the last one); for example, M, C (sequential), means that the BART model was first fine-tuned on the MS^2 dataset and then on the Cochrane dataset. All measures are obtained using the official evaluation script on the validation set.

lowing previous work (Huang et al., 2019). Specifically, we use classical unsupervised algorithms, namely TextRank (Mihalcea and Tarau, 2004) and LexRank (Erkan and Radev, 2004), that we chose since they are simple, well-studied and have a low computational cost. For each target, the extraction is performed on the whole pool of the concatenated abstracts. We also experiment with extracting sentences related to the results only from each abstract (which we then concatenate).

3.2.1 TextRank

TextRank constructs a graph using sentences as nodes and their similarity in terms of normalized number of words as edges. Then, the algorithm extracts the most central sentences according to PageRank (Page et al., 1999).

In order to extract the most important sentences only and minimize repetitions, we grouped all abstracts related to a single target and extracted the salient sentences from the whole pool of text. We used the summa library[3]; we constrained the summary obtained through TextRank to be approximately 1000 tokens (as this is the maximum number of tokens BART can process) and 500 tokens long (to experiment with even shorter salient inputs). Then, we fine-tuned a base BART model with the output data. Table 3 shows the results.

3.2.2 LexRank

Similarly to TextRank, LexRank constructs a graph using sentences as nodes and their similarity as edges; the similarity is computed in terms of term frequency-inverse document frequency (TF-IDF) vectors. Then most central sentences are extracted. We used the sumy[4] library for extraction and explored with outputs of a maximum of 30 sentences (as we estimate this will be compatible with BART's input constraint). Then, we fine-tuned a base BART model with the output data. Table 4 shows the results.

3.3 Extracting the abstracts' results to decrease the input size

Since a systematic review aims in assessing the knowledge in a given area, we explored extracting the results of each abstract only. To do so, we downloaded 150,000 random structured abstracts in English using the Pubmed Advanced Search Builder[5].

[3]https://github.com/summanlp/textrank

[4]https://github.com/miso-belica/sumy

[5]https://pubmed.ncbi.nlm.nih.gov/advanced/

Trained on	Eval on	R-1	R-2	R-L	BertScore	ΔEI avg	ΔEI macro
M - 1k tokens	M	12.7	1.15	9.79	83.02	51.98	43.32
C + M (mix) - 1k tokens	M	12.5	1.07	9.73	83.01	53.26	41.83
C + M (mix) - 500 tokens	M	13.21	1.3	10.13	83.24	49.87	42.87
C - 1k tokens	C	19.9	2.98	13.56	84.81	37.52	37.14
C + M (mix) - 1k tokens	C	22.63	6.09	16.95	86.89	31.92	38.83
M, C (sequential) - 1k tokens	C	19.47	3.4	13.75	84.94	36.63	38.43
C + M (mix) - 500 tokens	C	22.63	6.07	16.8	87	28.71	36.88

Table 3: Results obtained with a base BART model on inputs capped at around 1000 and 500 tokens extracted by TextRank algorithm. Some models are trained on the MS^2 dataset (M) or on the Cochrane dataset (C) independently. We also trained a single model with the mixed MS^2 and Cochrane data, in random order (mix) and evaluate it on both datasets independently. Finally, we experiment with sequential fine-tunings over the two datasets (with the fine-tuning over the target dataset being the last one); for example M, C (sequential), means that the BART model was first fine-tuned on the MS^2 dataset and then fine-tuned on the Cochrane dataset. All measures are obtained using the official evaluation script on the validation set.

Trained on	Eval on	R-1	R-2	R-L	BertScore	ΔEI avg	ΔEI macro
M	M	13.18	1.3	10.2	83.12	50.09	43.08
C + M (mix)	M	13.96	1.55	10.66	83.44	47.52	42.99
C	C	18.1	2.52	12.6	84.24	37.43	37.65
C + M (mix)	C	22.03	5.61	16.28	86.71	26.98	39.29

Table 4: Results obtained with a base BART model on inputs capped at around 30 sentences extracted by LexRank algorithm. Some models are trained on the M S2 dataset (M) or on the Cochrane dataset (C) independently. We also trained a single model with the mixed MS^2 and Cochrane data, in random order (mix) and evaluate it on both datasets independently. All measures are obtained using the official evaluation script on the validation set.

Trained on	Eval on	R-1	R-2	R-L	BertScore	ΔEI avg	ΔEI macro
M	M	12.97	1.27	10.02	83.09	49.59	42.48
C + M (mix)	M	12.61	1.61	9.69	82.96	52.36	41.98
C	C	22.42	5.84	16.59	86.82	30.05	38.02
C + M (mix)	C	22.95	6.17	16.9	86.94	28.43	36.97

Table 5: Results on the development set for the BART model after extracting the results only with a classifier. Some models are trained on the M S2 dataset (M) or on the Cochrane dataset (C) independently. We also trained a single model with the mixed MS^2 and Cochrane data, in random order (mix) and evaluate it on both datasets independently. All measures are obtained using the official evaluation script on the validation set.

Structured abstracts are divided into a number of sections with a related label (e.g., AIM, METHOD, CONCLUSIONS). We used regular expressions to divide the abstract into sections and extract the related label (we identified a label as a cased word or set of words at the start of a line followed by columns) and considered a section containing results as any section having as label CONCLUSION(S), CONCLUDING *, RESULT(S), SIGNIFICANCE, IMPORTANCE, RECOMMENDATION(S). We constructed a dataset assigning the positive label to sentences in such section and the negative label to sentences in the others. Since the negative instances were more than an order of magnitude more common than the positive ones, we balanced the dataset and obtained a sample of 700 negative sentences and 524 positive sentences. Then, we trained a Roberta base model to classify the sentences according to their labels. We used the dataset to extract sentences from the abstracts that have at least a 0.4 log prob of belonging to the positive class (we prefer to increase recall over accuracy, as the summarization step will remove pleonastic content). Then, we fine-tuned a BART base model with the concatenated results. Table 5 shows the obtained results.

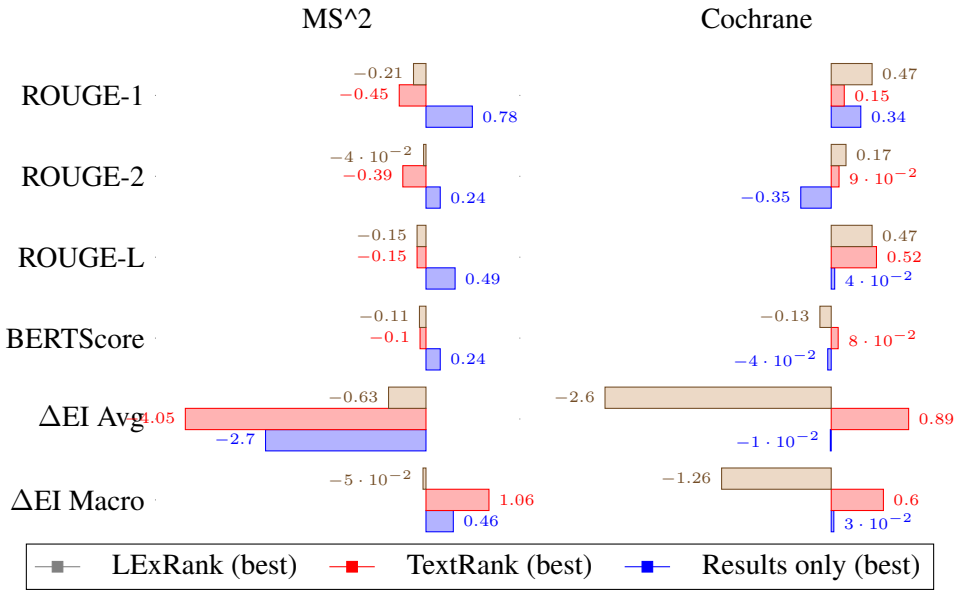

Figure 2: Effect of the extractive step. For each dataset, we consider the base model trained and evaluated on the target dataset as our baseline, and show the relative difference in performance when compared to the best model for each extractive algorithm. For LexRank, we considered the model trained on M+C mixed data, after extracting the salient sentences with LexRank. For TextRank, we considered the model trained on M+C mixed data, after extracting the salient sentences (500 tokens long for MS^2 and 1000 for Cochrane); for the Results only, we considered the model fine-tuned on MS^2 only for MS^2 and the mixed one for Cochrane.

4 Conclusions

We have explored a number of base BART models for the task of generating systematic reviews in the medical domain. Given the limited number of tokens BART can handle, we adopted several simple extractive strategies to retrieve salient sentences to the abstractive model; we also trained a model from the abstract results sentences only.

Generally, we found results on the Cochrane datasets are much more encouraging than those on the MS^2 and we believe that using the background info might improve performance. We found that the results obtained from the salient sentences only show mixed results. For MS^S, extracting the results sentences only seems to be the most promising method. For the Cochrane dataset, all extractive methods show small improvements over the baseline. LexRank seems to be the most promising, as it slightly improves the results, both in terms of ROUGE and factuality metrics.

In addition to ours, other strategies could be explored to sort the input abstract: DeYoung et al. (2021), for example, sorts abstracts by some measures of quality; it would be interesting to see how this compares to our proposed strategies. We also plan to explore different input representations that go beyond the simple concatenation of abstract and data augmentation techniques. Another possible route could be that of extracting domain-specific concepts, through, e.g., PubTator (Wei et al., 2013), to enrich abstracts.

References

Waleed Ammar, Dirk Groeneveld, Chandra Bhagavatula, Iz Beltagy, Miles Crawford, Doug Downey, Jason Dunkelberger, Ahmed Elgohary, Sergey Feldman, Vu Ha, Rodney Kinney, Sebastian Kohlmeier, Kyle Lo, Tyler Murray, Hsu-Han Ooi, Matthew Peters, Joanna Power, Sam Skjonsberg, Lucy Wang, Chris Wilhelm, Zheng Yuan, Madeleine van Zuylen, and Oren Etzioni. 2018. Construction of the literature graph in semantic scholar. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 84–91, New Orleans - Louisiana. Association for Computational Linguistics.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*.

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.

Jay DeYoung, Iz Beltagy, Madeleine van Zuylen, Bailey Kuehl, and Lucy Wang. 2021. MSˆ2: Multi-document summarization of medical studies. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7494–7513, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Günes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22(1):457–479.

Si Huang, Rui Wang, Qing Xie, Lin Li, and Yongjian Liu. 2019. An extraction-abstraction hybrid approach for long document summarization. *2019 6th International Conference on Behavioral, Economic and Socio-Cultural Computing (BESC)*, pages 1–6.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking : Bringing order to the web. In *WWW 1999*.

Byron C Wallace, Sayantan Saha, Frank Soboczenski, and Iain J Marshall. 2021. Generating (factual?) narrative summaries of RCTs: Experiments with neural multi-document summarization. *AMIA Summits Transl. Sci. Proc.*, 2021:605–614.

Chih-Hsuan Wei, Hung-Yu Kao, and Zhiyong Lu. 2013. Pubtator: a web-based text mining tool for assisting biocuration. *Nucleic acids research*, 41(W1):W518–W522.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.