ArXivSum: An Automatic and Representative Approach to Generating Summarization Datasets

Niv Eckhaus, Uri Stern, Shahar Spencer
{niv.eckhaus,uri.stern,shahar.spencer}@mail.huji.ac.il

1 Introduction

Summarization is one of the main downstream tasks in NLP, with important applications in many fields, such as medicine (Joshi et al., 2020) and academic research (Saggion et al., 2016). The popular benchmarks for summarization today are the XSum (Narayan et al., 2018) and CNN-DailyMail (Nallapati et al., 2016) datasets, which consist of news articles from 2007 to 2017. These datasets were generated by using either the first paragraph as the gold-standard label for the article's summary, or by finding the subset of sentences with the highest ROUGE score for the article, chosen among the first few sentences. Although this extractive method may work for News Articles, where the important content is generally located at the beginning of the text, for a general domain this method may not be sufficient. As the quality of the model is majorly dependent on the dataset it was trained and tested on, this is clearly problematic. Previous works (Gupta et al., 2021) have suggested methods to create improved datasets for the summarization task, using academic articles as the data source. The labels were annotated manually, making it difficult to create a large-scale dataset. We aim to suggest a method to automatically collect more data that would be more representative than the existing approach, yet can be generated simply and automatically at a large-scale. Our dataset is based on academic research papers taken from the ArXiv website. These articles contain abstracts that can be easily used as representative summaries of their contents.

2 Data

Our data was collected by scraping the ArXiv website. We wrote a script that receives a query describing the required domain and the number of requested samples. The script extracts their contents and stores them tagged under that domain. Overall, after dealing with technical blockings of our scraper by the ArXiv website, we managed to scrape 1686 article samples, taken from 3 varied domains under the field of Computer Science: Noisy Labels Identification, Verification and Routing Protocols. Full statistics are detailed in Table 1. These different domains were chosen to reflect a variety of disciplines, under the constraints of needing to have the same paper-format (in order to support our preliminary scraper). Since we were limited by the model's input tokens length, we considered settling for using the introduction of each article as its content, since it is the section readers tend to pay most attention to. This aligns with the introduction containing the fundamental parts of the article that should be in a summary: motivation, general description of the methods and a preface to the conclusions. After consulting with our advisor (Roy), we adopted this approach

3 Methods

In the evaluation of the dataset we created, we chose to focus on pre-trained models that were not fine-tuned on the summarization task, since we didn't want our model to be contaminated by the datasets mentioned above. Out of the Sequence-To-Sequence encoder-decoder Transformer models that are accessible, we chose T5 over BART. This derives from T5-Large's number of parameters (770M), which is bigger than BART-Large's (140M), and from T5's update that allows input of 1024 tokens (same as BART), in contrast to its previous limit of 512 tokens.

After collecting our data, we split each domain into a train, validation and test set. We modified a HuggingFace training script (HuggingFace, 2023) to fine-tune the model and to evaluate its performance with the ROUGE-Longest-Common-Sequence (ROUGE-L) metric (Lin, 2004). We used the standard Negative-Log-Likelihood loss function. All other hyperparameters remained the same as default in the most used summarization script on HuggingFace.

Our first step was training, validating and testing our model on each domain's subset separately. As the test set is similar to the train set, we expected the model to generate quality summaries. As a second step, we challenged the model by testing it on a test set from one of the other domains. We repeated the process until all combinations of training and test subsets were covered. This repetitive process

was done in order to get a normalized view of the results, and a better picture of our method's performance across different levels of required generalization. As a third step we also tested each trained model on a subset of the CNN-DM dataset's original test set. This subset was made by shuffling the test set and choosing 55 samples (to match the average size of the other domains' test sets). This step was done in order to examine the model's performance at a higher level of generalization, with a domain that is different than research papers.
We didn't want to settle for only using ROUGE-L as our quantitative metric, even though it is the standard evaluation method for summarization in popular benchmarks. It measures lexical similarity to a reference, rather than semantic similarity, which is majorly important in summarization. Therefore we chose to also evaluate using the newly available ChatGPT technology's API. This was done by giving ChatGPT the article's original introduction and the model's generated summary, and asking for two scores: *relevance* score, which determines whether the main points of the paper described in the introduction appear in the summary (following the logic of the precision score); and *alignment* score, which determines whether there are any details that appear in the summary but not in the introduction.

To make sure the scores are reasonable and not random, we used two different prompts (see Appendix): in the first one (6.1) we instructed ChatGPT to only output the scores, while in the second one (6.2) we used zero-shot Chain-of-Thought mechanism (Kojima et al., 2022), which also provides explanation for the scores (and is supposed to improve performance). Lastly, we ran many of the evaluations multiple times, to test the effect of ChatGPT's randomness on the scores. Notably, we saw that the two prompts were more often than not consistent with the output scores, seen in Tables 3 ,4. Almost all differences are below 10 percent, which is rather small and shows relative consistency, meaning the effect of ChatGPT's randomness is minor. There still were some instances of larger inconsistencies between the different prompts/runs. However, studying those cases goes beyond the scope of this project, and constitutes an interesting subject for future study.

4 Results

Regarding our ROUGE-L scores, as can be seen in Table 2, all of the train-test combinations of our data's domains resulted in scores in the range of 21.32-24.48. Their weighted-average is 22.61. This score is lower than the current state-of-the-art model's score over the CNN-DM dataset (Liu et al., 2022), 47.19. However, it is not that far away from ChatGPT's evaluated ROUGE-L score (Qin et al., 2023), 32.20, which might be the most widespread model, used by people outside the field of deep learning. An interesting point is that with more training data, the test results improve: the model that was trained on the domain with the largest train size (12% more samples on average) consistently scored highest across all test domains. In addition, the scores for all models on the CNN-DM test set were above 29.00, which is very close to ChatGPT's evaluated ROUGE score, and not that far from the state-of-the-art performance of a T5 variation on this dataset, 37.54 (Navarro et al., 2022).

As ROUGE-L score is inherently more suitable for extractive summarization, we also wanted to examine the results using ChatGPT, which we hoped would measure the semantic quality of the abstractive summaries, as mentioned before. The results were relatively high across the board: (Tables 3 ,4) all scores are between 80 and 95, with an average of *Relevance*: 88.3, *Alignment*: 86.8. We could not identify a training set with which the model performed particularly better, in contrast to the gap in the ROUGE scores. Consistently with the ROUGE evaluation, the CNN-DM's results were higher, which means the model managed to generalize on it. One explanation might be that News Articles are an "easier" type of data to understand, and therefore summarize, than complex computational articles.

In addition to these automatic evaluation methods, we evaluated the generated summaries manually, and saw that they were both coherent and non-repetitive.

5 Conclusions

We saw that our method indeed has the potential to create representative and large-scale datasets for summarization. Models trained on our data exhibited effective generalization across diverse domains and various document structures. Our assessment might indicate that increasing the size of the train set would improve the results, leaving opportunity for further research.

6 Appendix

6.1 First Prompt

I will now give you an introduction and a summary. these are an introduction of a scientific paper and an abstract proposal written based on this introduction. Please rate how well the abstract summarizes the article, based on the following parameters: 1. Relevance - whether the main points of the paper described in the introduction appear in the abstract. Provide a score between 0 (no details in the abstract) to 100 (all the introduction details are in the abstract). 2. Alignment - whether there are any details that appear in the abstract but not in the introduction. Provide a score between 0 (there are many details in the abstract that do not appear in the introduction) to 100 (there are no extra details in the abstract). Your output should only be the score in exactly this format: (Relevance score,Alignment score). there should be one such score for the abstract. The introduction: {introduction} The abstract: {abstract}

6.2 Zero-Shot Chain of Thought Prompt

I will now give you an introduction and a summary. these are an introduction of a scientific paper and an abstract proposal written based on this introduction. Please rate how well the abstract summarizes the article, based on the following parameters: 1. Relevance - whether the main points of the paper described in the introduction appear in the abstract. Provide a score between 0 (no details in the abstract) to 100 (all the introduction details are in the abstract). 2. Alignment - whether there are any details that appear in the abstract but not in the introduction. Provide a score between 0 (there are many details in the abstract that do not appear in the introduction) to 100 (there are no extra details in the abstract). there should be one such score for the abstract. Let's think step by step, and conclude with the scores in exactly this format: (Relevance score,Alignment score). The introduction: {introduction} The abstract: {abstract}

6.3 Link to Our GitHub Repository

https://github.com/shaharspencer/ANLP_group_project

Domain	Train	Val	Test	Sum
NL	430	124	62	616
V	395	115	56	566
RP	354	99	51	504
Overall	1179	338	169	1686

Table 1: Data statistics. *Train*, *Val* and *Test* are the number of samples in each split (70%, 20% and 10% respectively), and *Sum* is their sum. Domains: Noisy Labels Identification (*NL*), Verification (*V*) and Routing Protocols (*RP*).

Train\Test	NL	V	RP	CD
NL	23.82	24.48	21.73	29.61
V	22.16	23.23	21.32	29.89
RP	22.26	22.50	21.69	29.05

Table 2: Reports of ROUGE-L scores for various train/test splits. Rows describe the domain of the train split, columns describe the domain of the test split. Domains initials are as in Table 1, in addition to the CNN-DailyMail (*CD*) test set.

Train\Test	NL	V	RP	CD
NL	83.0, 84.5	85.5, 79.5	79.5, 70.0	88.7, 86.2
V	75.5, 82.0	82.5, 81.0	81.0, 81.5	86.9, 87.8
RP	82.5, 86.0	79.0, 78.0	81.5, 81.5	89.4, 84.6

Table 3: Reports of ChatGPT scores (*relevance, alignment*), using the first prompt, for various train/test splits. Rows, columns and domains are as describe in Table 2.

Train\Test	NL	V	RP	CD
NL	79.9, 76.3	83.2, 83.5	74.5, 79.0	85.6, 87.1
V	76.7, 74.0	81.0, 85.3	87.0, 88.5	91.3, 86.4
RP	85.7, 87.3	81.0, 89.0	76.0, 79.0	87.9, 88.5

Table 4: Reports of ChatGPT scores (*relevance, alignment*), using the second prompt (with zero-shot chain of thought), for various train/test splits. Rows, columns and domains are as describe in Table 2.

References

Vivek Gupta, Prerna Bharti, Pegah Nokhiz, and Harish Karnick. 2021. SumPubMed: Summarization dataset of PubMed scientific articles. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*, pages 292–303, Online. Association for Computational Linguistics.

HuggingFace. 2023. Huggingface transformers summarization. https://github.com/huggingface/transformers/tree/main/examples/pytorch/summarization.

Anirudh Joshi, Namit Katariya, Xavier Amatriain, and Anitha Kannan. 2020. Dr. summarize: Global summarization of medical dialogue by exploiting local structures. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3755–3763, Online. Association for Computational Linguistics.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

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.

Yixin Liu, Pengfei Liu, Dragomir R. Radev, and Graham Neubig. 2022. Brio: Bringing order to abstractive summarization. In *Annual Meeting of the Association for Computational Linguistics*.

Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, Çaglar Gülçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Conference on Computational Natural Language Learning*.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

David Fraile Navarro, Mark Dras, and Shlomo Berkovsky. 2022. Few-shot fine-tuning sota summarization models for medical dialogues. In *North American Chapter of the Association for Computational Linguistics*.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is chatgpt a general-purpose natural language processing task solver? *ArXiv*, abs/2302.06476.

Horacio Saggion, Ahmed Ghassan Tawfiq AbuRa'ed, and Francesco Ronzano. 2016. Trainable citation-enhanced summarization of scientific articles. In *BIRNDL@JCDL*.