CSIRO Data61 Team at BioLaySumm Task 1: Lay Summarisation of Biomedical Research Articles Using Generative Models

Mong Yuan Sim[1,2*] **Xiang Dai**[1] **Maciej Rybinski**[1] **Sarvnaz Karimi**[1]
[1]CSIRO Data61 [2]The University of Adelaide
mongyuan.sim@student.adelaide.edu.au
{dai.dai;maciek.rybinski;sarvnaz.karimi}@csiro.au

Abstract

Lay summarisation aims at generating a summary for a non-expert audience which allows them to keep updated with the latest research in a specific field. Despite the significant advancements made in the field of text summarisation, lay summarisation remains relatively under-explored. We present a comprehensive set of experiments and analyses to investigate the effectiveness of existing pre-trained language models in generating lay summaries, focusing on the impact of two factors: model size and training data. When evaluating our models in BioLaySumm Shared Task, our submission ranked second for the relevance criteria and third overall among 21 competing teams.[1]

1 Introduction

Text summarisation (Spärck Jones, 1999) is a task in Natural Language Processing (NLP) where its goal is to generate a short and informative summary. Recent developments of summarisation techniques are supported by advancement in pre-trained language models (Devlin et al., 2019; Lewis et al., 2020; Raffel et al., 2020) and availability of large-scale datasets which include news articles (Fabbri et al., 2019), scientific publications (Lu et al., 2020) and meeting records (Zhong et al., 2021).

Despite the advances in generic text summarisation, lay summarisation for scientific documents remains less explored. That is, there has been less focus on simplifying technical terms in scientific documents, causing the generated summary to have the same readability level as the input text. Lay summarisation aims to produce a non-technical summary for technical articles (Goldsack et al., 2022). For instance, a lay summary for a scientific article should not contain any technical terms and should be readable and understandable to the public. The lay summary is especially important in the biomedical area, where people with no relevant background such as journalists, interdisciplinary researchers, or patients may want to keep updated with recent advances in the field.

We participate in the BioLaySumm shared task (Goldsack et al., 2023) to investigate methods of generating high-quality lay summaries for scientific articles. We present our empirical findings from three different perspectives: (1) model size; (2) data augmentation techniques; and, (3) input length. Our contributions are two-fold: (1) We explore the effectiveness of using pre-trained language models for fine-tuning and data augmentation in lay summarisation, and (2) We investigate the impact of input length on the quality of the generated summary. In the final shared task rankings,[2] our approach ranks second for relevance metric and third overall among 21 teams, with a relatively high balance across all metrics.

2 Related Work

Text Summarisation aims to produce a short and concise summary that is representative of information included in input text (Spärck Jones, 1999; Ma et al., 2022a). There are two different types of summarisation: *extractive* and *abstractive*. The extractive summarisation model selects important sentences from input text while the abstractive summarisation model has the ability to generate a summary that contains words that do not exist in the input text. There have been promising advances in generating text summary using pre-trained language models (Xiao et al., 2022; Ma et al., 2022b), graph-based summarisation (Yasunaga et al., 2017; Liao et al., 2018; Li et al., 2020; Pasunuru et al., 2021), and hierarchical models (Fabbri et al., 2019; Liu and Lapata, 2019a).

Pre-trained LMs in Summarisation Text summarisation has benefited from pre-trained language

*This work was partially done when Sim was a summer intern at CSIRO Data61.

[1]https://github.com/raymondsim/biolaysumm

[2]https://biolaysumm.org/

The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, pages 629–635

models such as BERT (Devlin et al., 2019), BART (Lewis et al., 2020), and T5 (Raffel et al., 2020). Most studies however used pre-training objectives that are not related to summary generation. Summarisation-specific pre-trained language models are also studied. Liu and Lapata (2019b) introduced BERTSUM, which is capable of generating both extractive and abstractive summaries with a document-level encoder built on top of BERT. Zhang et al. (2020a) proposed PEGASUS, a transformer-based pre-trained language model where summary generation is explicitly included as a pre-training objective. The model is trained to generate abstractive summaries from masked sentences in a document. Here, we leverage the pre-trained LMs not only to generate lay summary but also to generate additional training data.

3 Approach

Backbone Models T5 (Raffel et al., 2020) is a transformer-based text-to-text pre-trained language model. Different tasks, including translation, summarisation, question answering, and classification, are all converted into a text-to-text format. FLAN-T5 (Chung et al., 2022) is an enhanced version of T5 using instruction fine-tuning. That is, the original T5 model is fine-tuned on tasks phrased as instructions.

We use pre-trained FLAN-T5 models of different sizes (i.e., base, large, and xl),[3] with the same instruction across all experiments: "summarise the following article: [DOCUMENT] Summary:". Since the model should generate long summaries, we use a beam search decoder during inference—a beam width of four—to generate up to 512 tokens. Table 1 shows the training time for fine-tuning these models.

Intermediate-task Pretraining Due to the relatively small size of eLife training set, we also explore the effectiveness of intermediate-task pretraining (Pruksachatkun et al., 2020; Rybinski et al., 2021; Yu et al., 2021) to improve the effectiveness of fine-tuned FLAN-T5 models. Specifically, we first fine-tune FLAN-T5 on a target task-related data set (e.g., PLOS training set and XSum (Narayan et al., 2018) and then on the eLife training set. In addition, we employ an off-the-shelf pre-trained language model (i.e., GPT-3.5) to generate additional training data, which will be explained in detail in Section 5.

[3] https://huggingface.co/docs/transformers/model_doc/flan-t5

	# parameters	eLife	PLOS
FLAN-T5-base	250M	5.1	5.2
FLAN-T5-large	780M	14.6	14.8
FLAN-T5-xl	3B	43.1	41.9

Table 1: Training time (GPU-hours) for fine-tuning FLAN-T5 models on eLife and PLOS training data. We run all experiments on a cluster consisting of Tesla V100 (32GB GPU memory).

4 Datasets and Evaluation Metrics

There are two lay summarisation datasets used for the shared task, namely eLife (Goldsack et al., 2022) and PLOS (Goldsack et al., 2022; Luo et al., 2022). eLife[4] is a scientific journal for biomedical and life sciences. Publications on eLife contain lay summaries written by expert editors based on the journal article to explain the background and key points of a scientific article to the non-expert. PLOS[5] (Public Library of Science) is an open-access journal, focusing on science and medicine publications where each publication is paired with a lay summary provided by the authors themselves.

The shared task organisers provided training and development sets for model development, and two hidden test sets to score submissions. The descriptive statistics of these datasets can be found in Table 2.

We follow the setup of the shared task to evaluate generated summaries across three aspects: Relevance (ROUGE, BERTScore), Readability (FKGL, DCRS), and Factuality (BARTScore). The aim is to maximise the scores for Relevance and Factuality metrics and minimise scores for Readability metrics. We provide a short description of these metrics and refer readers to (Goldsack et al., 2023) for more details.

[4] https://elifesciences.org/
[5] https://plos.org/

	eLife			PLOS		
	train	dev	test	train	dev	test
# examples	4346	241	142	24773	1376	142
Avg article nt.	17.3K	17.0K	15.3K	10.9K	10.8K	11.0K
Avg summary nt.	521	528	—	288	288	—

Table 2: Descriptive statistics of the datasets. nt.: number of tokens, measured using the FLAN-T5 tokenizer. The test summaries have not been released yet.

Dataset	Model	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	FKGL	DCRS	BARTScore
eLife Dev	FLAN-T5-base	0.411	0.108	0.393	0.834	10.148	**7.100**	**-2.101**
	FLAN-T5-large	0.469	0.136	0.446	0.849	**9.700**	7.850	-2.213
	FLAN-T5-xl	**0.495**	**0.146**	**0.469**	**0.855**	9.783	8.170	-2.406
PLOS Dev	FLAN-T5-base	0.493	0.186	0.455	0.863	15.132	**11.088**	**-1.882**
	FLAN-T5-large	0.497	0.187	0.459	0.864	14.948	11.163	-1.894
	FLAN-T5-xl	**0.502**	**0.190**	**0.462**	**0.865**	**14.826**	11.194	-1.908
eLife Test	FLAN-T5-xl	0.480	0.130	0.454	0.854	9.804	8.224	-2.493
PLOS Test ‡	FLAN-T5-xl	0.497	0.194	0.460	0.867	15.089	11.372	-1.887

Table 3: The impact of model size on effectiveness. Base model has 250M parameters; large has 780M parameters; and, xl has 3B parameters. ‡ indicates our final submission.

ROUGE (Lin, 2004) assesses the quality of generated summaries by comparing generated summaries to gold summaries and counting the number of overlapping n-grams, word sequences, and word pairs between summaries.

BERTScore (Zhang et al., 2020b) uses contextual embedding in BERT (Devlin et al., 2019) to compute the similarity score for each token between the source document and generated summary and average them to obtain the overall effectiveness metric of a summarisation model.

FKGL (Flesch-Kincaid Grade Level) (Flesch, 1948) indicates difficulty in reading a passage in English based on two factors: the average sentence length and the average number of syllables per word.

DCRS (Dale-Chall Readability Score) (Chall and Dale, 1995) is a score based on occurrence of words unknown to most 4th-grade students (in the US education system). A lower score indicates higher readability.

BARTScore (Yuan et al., 2021) defines the evaluation of generated text as a text generation problem that uses BART as its backbone. It utilises the generation probabilities of BART to measure the quality of a sentence.

5 Experimental Results

Impact of Scaling the Model Size

We empirically compare the effectiveness of three publicly available pre-trained FLAN-T5 models which differ only in the number of parameters: FLAN-T5-base (250M), FLAN-T5-large (780M), and FLAN-T5-xl (3B). We fine-tune these FLAN-T5 models on the eLife—for 25 epochs—and PLOS—for 5 epochs—training sets respectively.[6] The fine-tuned models are then evaluated on the corresponding development sets.

Table 3 shows that larger models always result in better relevance scores (i.e., ROUGE scores and BERTScore), but not necessarily on readability scores (i.e., FKGL and DCRS) or factuality score (i.e., BARTScore). In fact, the smallest model—FLAN-T5-base—achieves the best BARTScore and DCRS scores, when evaluating eLife and PLOS development sets.

We use FLAN-T5-xl in the following experiments as default, unless otherwise specified.

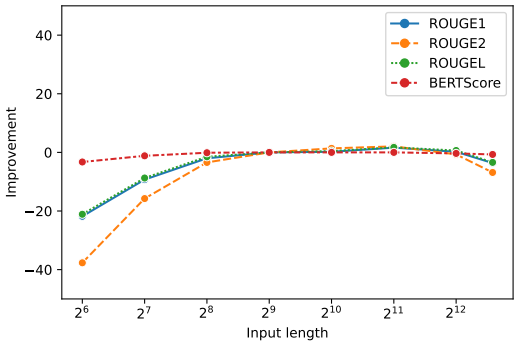
GPT-3.5 to Generate Additional Training Data

One observation from Table 3 is that evaluation scores on the eLife sets fall behind those on the PLOS sets. We hypothesise that this may be attributed to either the different characteristics of these two datasets (Table 2) or the relatively small size of the eLife training set (4346 examples vs. 24773 examples in PLOS training set). In order to understand the trade-off between domain and training data size, we conduct a cross-dataset evaluation. That is, we evaluate the effectiveness of the model that is trained using the PLOS training set, on the eLife development set. In addition, we use the combination of the eLife and PLOS training sets to train the model, which is then evaluated on the eLife development set. Results in the upper part of Table 4 show that although the PLOS set and the combined set contain a much larger number of training examples than the eLife training set, the models trained on the former two under-perform compared to the model trained only on eLife by a large margin (all three models are evaluated on the eLife development set).

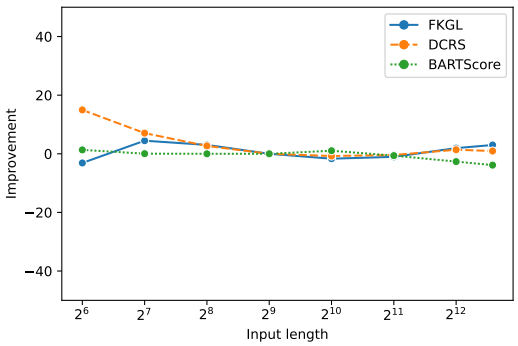
[6] Because the eLife training set is smaller than PLOS, we choose a larger number of training epochs for it.

	#	ROUGE-1	ROUGE-2	ROUGE-L	BERTS.	FKGL	DCRS	BARTS.
				eLife Dev				
eLife	4346	0.495	0.146	0.469	0.855	**9.783**	**8.170**	-2.406
PLOS	24773	0.342	0.075	0.314	0.835	15.364	11.553	-2.421
Combined	29119	0.449	0.126	0.426	0.846	10.334	8.175	**-2.223**
PLOS → eLife	24773 + 4346	0.503	**0.152**	0.478	0.856	9.918	8.237	-2.415
XSum → eLife	203017 + 4346	0.495	0.147	0.470	0.855	9.866	8.176	-2.398
GPT-3.5 (P) → eLife	4346 + 4346	**0.510**	**0.152**	**0.484**	**0.857**	9.904	8.196	-2.528
GPT-3.5 (S) → eLife	4346 + 4346	0.502	0.151	0.477	0.856	10.052	8.215	-2.439
				eLife Test				
GPT-3.5 (P) → eLife‡	4346 + 4346	0.489	0.130	0.463	0.855	10.013	8.316	-2.612

Table 4: The impact of training data on the effectiveness. $A \rightarrow B$ indicates sequential transfer learning where FLAN-T5-xl model is fine-tuned on A training data and then on B. # shows the number of training examples.. ‡ indicates our final submission.



(a) Relevance metrics



(b) Readability and factuality metrics

Figure 1: The impact of input sequence length on evaluation scores on the eLife development set. We use the input length of 512 (i.e., 2^9) as the baseline and measure the relative improvement due to different input lengths.

We also investigate whether intermediate-task pretraining can be used to improve the model training when only a relatively small size of training data (i.e., eLife) is available. More specifically, we first fine-tune FLAN-T5-xl on a source summarisation dataset (*source*), and then continue fine-tuning the model on the eLife training set (*target*). We consider four possible sources: (1) PLOS, which focuses on the task of lay summarisation with data sampled from different journals; (2) XSum (Narayan et al., 2018), which focuses on news articles summarisation; we choose it because of its relatively large training data size and simpler language used in its summaries; (3) GPT-3.5 (P), where for each target training example, a paraphrased summary is generated using OpenAI API (gpt-3.5-turbo, `https://platform.openai.com/docs/models/gpt-3-5`);[7] and, (4) GPT-3.5 (S), where GPT-3.5 is asked to generate summary for each document in the target training set.[8] Results in Table 4 show that all sources can help the model with improved relevance scores (ROUGE and BERTScore), demonstrating the benefits of using additional training data. On one hand, GPT-3.5 (P) outperforms other sources in all four relevance metrics, although its size is much smaller than PLOS and XSum training set. On the other hand, this benefit becomes unclear regarding the readability and the factuality metrics. For example, the model effectiveness in terms of readability scores (FKGL and DCRS) decreases for all the sources, and the model trained on the combination of eLife and PLOS achieves the highest BARTScore (factuality), whereas other models achieve very close results. Considering the low cost of using GPT-3.5 to generate synthetic data, it is a promising direction to use it for data augmentation and even for data annotation (Wang et al., 2022).

[7]Prompt template: "Paraphrase the following paragraph: [Summary] Paraphrased:"

[8]Prompt template: "Summarise the following document using plain text: [Input document] Summary:"

Impact of Input Document Length on Generated Summary

We use the input sequence length of 512 as a baseline and measure the relative improvement or decline of different scores as we change to different input lengths of test examples. Figure 1a shows that the model fails to generate a relevant summary (decreased ROUGE scores) if the input document is truncated to a shorter length. In contrast, taking a longer document as input to the summarisation model slightly improves the relevance scores. However, they decrease again when the input length is more than 2000 tokens, which potentially highlights the difficulty of capturing long-range contextual dependencies. Results on readability and factuality metrics (Figure 1b) show that these metrics are less affected by the length of the input document. In fact, the model can generate fluent text—with good BARTScore DCRS and scores—even with just a few tokens provided as input.

6 Conclusions

We present our approach to the lay summarisation for scientific documents in BioNLP 2023 shared task. Our approach utilises GPT-3.5 to generate additional training data and pre-trained FLAN-T5 in generating a lay summary for scientific documents. Our results show that extra data generated from a generative model can boost the effectiveness of a summarisation model to a certain degree, especially in terms of relevance metrics. In addition, we showed that input length impacts the relevance of generated summary, but has no obvious impact on the readability and factuality metrics. Future work could focus on developing a summarisation model that optimises readability and factuality of generated summary.

Limitations

In this work, we used GPT-3.5 to generate additional training examples and showed that it helps improve the relevance (ROUGE and BERTScore) of the generated summary. However, we did not explicitly analyse the quality of generated examples to check whether or not they are faithful and factually correct which could lead to the same problem in generated summaries. To obtain better training examples, we could use faithful or factuality metrics to assess generated training examples and then use the post-editing method or human evaluation to remove unfaithful content, which we leave for future research.

Acknowledgements

This work is supported by The Commonwealth Scientific and Industrial Research Organisation (CSIRO) Precision Health Future Science Platform (FSP). Experiments were undertaken with the assistance of resources and services from the National Computational Infrastructure (NCI), which is supported by the Australian Government.

References

J.S. Chall and E. Dale. 1995. *Readability Revisited: The New Dale-Chall Readability Formula*. Brookline Books.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

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.

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy. Association for Computational Linguistics.

Rudolph Flesch. 1948. A new readability yardstick. *Journal of Applied Psychology*, 32(3):221–233.

Tomas Goldsack, Zheheng Luo, Qianqian Xie, Carolina Scarton, Matthew Shardlow, Sophia Ananiadou, and Chenghua Lin. 2023. Overview of the biolaysumm 2023 shared task on lay summarization of biomedical research articles. In *Proceedings of the 22st Workshop on Biomedical Language Processing*, Toronto, Canada. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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.

Wei Li, Xinyan Xiao, Jiachen Liu, Hua Wu, Haifeng Wang, and Junping Du. 2020. Leveraging graph to improve abstractive multi-document summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6232–6243, Online. Association for Computational Linguistics.

Kexin Liao, Logan Lebanoff, and Fei Liu. 2018. Abstract Meaning Representation for multi-document summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1178–1190, Santa Fe, New Mexico, USA. 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.

Yang Liu and Mirella Lapata. 2019a. Hierarchical transformers for multi-document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5070–5081, Florence, Italy. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019b. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Yao Lu, Yue Dong, and Laurent Charlin. 2020. Multi-XScience: A large-scale dataset for extreme multi-document summarization of scientific articles. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8068–8074, Online. Association for Computational Linguistics.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2022. Readability controllable biomedical document summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4667–4680, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Congbo Ma, Wei Emma Zhang, Mingyu Guo, Hu Wang, and Quan Z. Sheng. 2022a. Multi-document summarization via deep learning techniques: A survey. *ACM Comput. Surv.*, 55(5).

Congbo Ma, Wei Emma Zhang, Mingyu Guo, Hu Wang, and Quan Z Sheng. 2022b. Multi-document summarization via deep learning techniques: A survey. *ACM Computing Surveys*, 55(5):1–37.

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.

Ramakanth Pasunuru, Mengwen Liu, Mohit Bansal, Sujith Ravi, and Markus Dreyer. 2021. Efficiently summarizing text and graph encodings of multi-document clusters. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4768–4779, Online. Association for Computational Linguistics.

Yada Pruksachatkun, Jason Phang, Haokun Liu, Phu Mon Htut, Xiaoyi Zhang, Richard Yuanzhe Pang, Clara Vania, Katharina Kann, and Samuel R Bowman. 2020. Intermediate-Task Transfer Learning with Pretrained Models for Natural Language Understanding: When and Why Does It Work? In *ACL*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Maciej Rybinski, Xiang Dai, Sonit Singh, Sarvnaz Karimi, and Anthony Nguyen. 2021. Extracting Family History Information From Electronic Health Records: Natural Language Processing Analysis. *JMIR Med Inform*, 9.

Karen Spärck Jones. 1999. Automatic summarizing: factors and directions. *Advances in automatic text summarization*, pages 1–12.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*.

Wen Xiao, Iz Beltagy, Giuseppe Carenini, and Arman Cohan. 2022. PRIMERA: Pyramid-based masked sentence pre-training for multi-document summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5245–5263, Dublin, Ireland. Association for Computational Linguistics.

Michihiro Yasunaga, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir Radev. 2017. Graph-based neural multi-document summarization. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 452–462, Vancouver, Canada. Association for Computational Linguistics.

Tiezheng Yu, Zihan Liu, and Pascale Fung. 2021. AdaptSum: Towards Low-Resource Domain Adaptation for Abstractive Summarization. In *NAACL*.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. In *Advances in Neural Information Processing Systems*, volume 34, pages 27263–27277. Curran Associates, Inc.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

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

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, and Dragomir Radev. 2021. QMSum: A new benchmark for query-based multi-domain meeting summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5905–5921, Online. Association for Computational Linguistics.