UIUC_BioNLP at BioLaySumm: An Extract-then-Summarize Approach Augmented with Wikipedia Knowledge for Biomedical Lay Summarization

Zhiwen You[1], Shruthan Radhakrishna[2], Shufan Ming[1], Halil Kilicoglu[1]
[1]School of Information Sciences, University of Illinois Urbana-Champaign, USA
[2]Department of Computer Science, University of Illinois Urbana-Champaign, USA
{zhiweny2, sr73, shufanm2, halil}@illinois.edu

Abstract

As the number of scientific publications is growing at a rapid pace, it is difficult for laypeople to keep track of and understand the latest scientific advances, especially in the biomedical domain. While the summarization of scientific publications has been widely studied, research on summarization targeting laypeople has remained scarce. In this study, considering the lengthy input of biomedical articles, we have developed a lay summarization system through an extract-then-summarize framework with large language models (LLMs) to summarize biomedical articles for laypeople. Using a fine-tuned GPT-3.5 model, our approach achieves the highest overall ranking and demonstrates the best relevance performance in the BioLaySumm 2024 shared task[1].

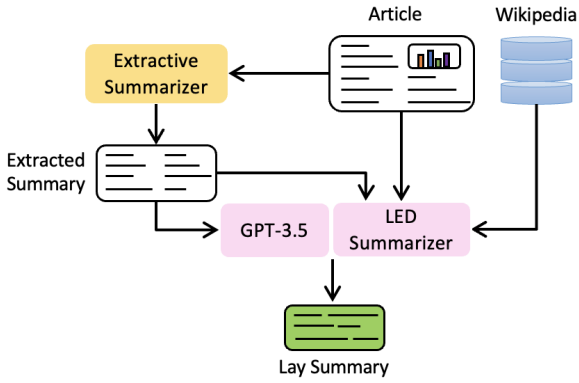

Figure 1: Our Extract-then-Summarize framework for the biomedical lay summarization task. We assess the performance of two models, GPT-3.5 and LED, in generating lay summaries. The input of the LED model includes the article sections that are ranked for relevance, Wikipedia knowledge, and an extractive summary. Meanwhile, the GPT-3.5 model is fine-tuned by the extractive summaries.

1 Introduction

New research in the biomedical field is often reported in the latest scientific articles and plays a crucial role in improving human health and well-being. However, the complex terminology and scientific language used in these publications can be challenging to understand for those without extensive knowledge of the field. The Biomedical Lay Summarization task (BioLaySumm) (Goldsack et al., 2024) addresses this challenge by creating summaries that are easier to read and understand, specifically tailored for readers unfamiliar with biomedical studies. Unlike traditional text summarization, which aims to condense documents into brief summaries, BioLaySumm also focuses on using easy-to-understand language. This approach ensures that the summaries are less technical and more accessible, while traditional summarization tasks prioritize capturing the precise scientific terminology used in the original documents. To solve the limited size and scope issue in current lay summarization corpora, Goldsack et al. (2022) have proposed two novel lay summarization datasets, PLOS and eLife, including biomedical journal articles alongside expert- and author-written lay summaries, which form the basis for the shared task.

We submit lay summaries generated by two fine-tuned LLMs for this task (one for each dataset). Considering the constraints on input size and computational resources (i.e., one GPU with 32 GB memory), using the full length of scientific articles for LLM fine-tuning is not feasible. Therefore, we adopt an extract-then-summarize approach (Koh et al., 2022; Bajaj et al., 2021) (illustrated in Figure 1), which allows us to reduce the input length while maintaining competitive performance. We fine-tune the Longformer Encoder-Decoder (LED) model (Beltagy et al., 2020) and GPT-3.5 (OpenAI, 2024) and explore the effectiveness of combining unsupervised extractive summarization methods with a retrieval-augmented gener-

[1]Our code is available at https://github.com/zhiwenyou103/UIUC_BioNLP_BioLaySumm2024.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 132–143

ation (RAG) approach (Guo et al., 2024) in generating lay summaries. Our experimental results on GPT-3.5 achieve the best overall ranking and the highest relevance score in the shared task.

2 Methods

We introduce our methodology for the biomedical lay summarization task (illustrated in Figure 1), including dataset description, section re-ranking, extractive summarization, RAG, GPT-3.5 fine-tuning, and evaluation measures. The detailed experimental settings are reported in Appendix A.

2.1 Datasets

We use eLife and PLOS datasets provided by Goldsack et al. (2022) for experiments. We report the average tokens of each dataset given the whole document and lay summarization in Appendix B. The lay summaries of eLife are crafted by expert editors, offering extensive abstraction and enhanced readability. Conversely, PLOS presents lay summaries written directly by the authors of articles. In terms of articles, eLife comprises peer-reviewed publications encompassing a broad spectrum of life sciences and medicine. PLOS covers journals spanning Biology, Computational Biology, Genetics, Pathogens, and Neglected Tropical Diseases (Goldsack et al., 2022).

2.2 Preprocessing

We employ two methods to preprocess the input article, aiming to reduce its length and extract salient sentences: section reordering and unsupervised extractive summarization.

2.2.1 Section Reordering

To better understand the experimental datasets, we conduct preliminary experiments to analyze which sections are most relevant to the gold standard lay summaries. We first group the headings of each article in eLife and PLOS datasets into five categories using structured section labels provided by the National Library of Medicine (NLM)[2] (See Appendix C for more details). Then, based on the results of section-level similarity comparison, we reorder the whole article in the order of abstract, background, conclusion, results, and methods sections. The results of the reordering method in the eLife validation set in Appendix C show the effectiveness of the restructured article compared with the default section order.

2.2.2 Unsupervised Extractive Summarization

Given the input length constraints and limited computing resources, fully incorporating scientific articles for model fine-tuning is impractical. To capture the essential global information of the articles, we implement two unsupervised extractive summarization approaches: a graph-based ranking method (TextRank) (Mihalcea and Tarau, 2004) and a BERT-based clustering method (Miller, 2019), to extract salient sentences from the documents.

TextRank (Mihalcea and Tarau, 2004) operates by treating text as a graph, where nodes are constructed based on lemmas, parts-of-speech tags of tokens in the text, and edges based on co-occurrence within a window. By iteratively applying a ranking algorithm similar to Google's PageRank (Brin and Page, 1998), it identifies the essential tokens, helping generate summaries or extract key information from text documents.

We use a BERT-based clustering approach (Miller, 2019) for unsupervised extractive summarization. It starts by dividing the article into segments using the LangChain[3] `NLTKTextSplitter` API. Next, we apply a pre-trained embedding model PubMedBERT[4] $\mathcal{E}$ deployed through SentenceTransformers[5] to encode sentences from both lay summaries and segmented passages. We calculate the cosine similarity between these embeddings to create a contrastive learning dataset $\mathcal{C}$, essential for fine-tuning the embedding model adapted for lay summarization tasks. Specifically, pairs with cosine similarity scores above 0.9 are considered positive, indicating high relevance, while those with scores below 0.01 are negative, indicating minimal relevance. We then fine-tune $\mathcal{E}$ with $\mathcal{C}$ through a contrastive loss. Appendix B presents the created dataset statistics of $\mathcal{C}$. Following the method proposed by Miller (2019), we apply a K-means clustering approach to group sentences with the same themes and find the sentences closest to the cluster's centroids as salient sentences. We extract the top 50 closest sentences from all clusters, each with a maximum length of 256

[2] https://lhncbc.nlm.nih.gov/ii/areas/structured-abstracts/downloads/Structured-Abstracts-Labels-102615.txt

[3] https://www.langchain.com/

[4] https://huggingface.co/NeuML/pubmedbert-base-embeddings

[5] https://huggingface.co/sentence-transformers

Models	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
BART (baseline)	0.4696	0.1395	0.4358	0.8623	12.0359	10.1475	13.4852	48.0963	0.7788	0.7026
GPT-3.5	0.4855	**0.1569**	0.455	**0.8677**	**11.7535**	**9.3388**	**13.3642**	52.8504	**0.8004**	**0.7338**
PubMed$_{LED}$	**0.4926**	0.1563	**0.4576**	0.8585	12.4500	9.8969	13.4096	**63.7736**	0.7576	0.6828

Table 1: Performance of our final submission and the baseline models on the test sets of both eLife and PLOS datasets. BART represents the baseline model proposed by the BioLaySumm organisers. GPT-3.5 is our final submission on the leaderboard, and PubMed$_{\text{LED}}$ is an open-source model for comparison. PubMed$_{\text{LED}}$ model tuning involves reordered sections, extractive summary, and RAG (i.e., DPR and Wikipedia definition retrieval).

tokens, as an extractive summary.

These two extractive summarization approaches reduce the overall document length, capture the essential global context, and facilitate efficient model fine-tuning.

2.3 Retrieval-Augmented Generation

To simplify model-generated lay summaries, we resort to external knowledge due to the limited background information in the datasets. First, we use a keyword-based definition retrieval method to extract definitions from the Wikipedia dataset (Guo et al., 2024) through string matching. Specifically, we employ KeyBERT[6] to extract the top 10 keywords from each article's abstract using BERT embeddings. Then, we use dataset-provided and extracted keywords to retrieve short definitions from the Wikidata-based dataset (Guo et al., 2024). We use the Wikipedia API via LangChain for extended definitions if no results are found. We concatenate the retrieved information with the input article.

Additionally, we apply an embedding-based method to extract relevant information by selecting passages from the "wiki_dpr" dataset, which contains 21 million 100-word passages from Wikipedia (Lewis et al., 2020b). Using the pre-trained dense retrieval (DPR) component of the RAG model, we retrieve the five most relevant passages to integrate into our generation tasks.

2.4 GPT-3.5 Fine-Tuning

We also experiment with GPT-3.5-turbo[7], a large-scale closed-source model from OpenAI. Our experiments, along with findings from Turbitt et al. (2023), demonstrate performance below the baseline in zero-shot and few-shot prompting settings. Consequently, we investigate fine-tuning the model using the OpenAI API[8]. To minimize API costs, we employ the extract-then-summarize approach, utilizing TextRank to extract key sentences from the full text, which are then fed into the GPT model for summary generation. Our results indicate that fine-tuning on small random samples (100 to 400 examples) is adequate to achieve high performance for the task. For our final submission, we extract 40 sentences per article using TextRank and fine-tune separate models for each dataset using random samples of 200 articles.

2.5 Evaluation

We assess the performance of our model using the official evaluation scripts provided by the organizers (Goldsack et al., 2023), employing various automatic metrics related to relevance, readability, and factuality. Relevance is measured through ROUGE (1, 2, and L) (Lin, 2004) and BERTScore (Zhang et al.). Readability metrics include the Flesch-Kincaid Grade Level (FKGL) (Kincaid et al., 1975), Dale-Chall Readability Score (DCRS) (Dale and Chall, 1948), Coleman-Liau Index (CLI) (Coleman and Liau, 1975), and LENS (Maddela et al., 2023). Notably, lower FKGL, DCRS, and CLI scores signify improved readability. Factuality evaluation incorporates AlignScore (Zha et al., 2023) and SummaC (Laban et al., 2022).

3 Results and Analysis

We present the evaluation results of our methods in the leaderboard in Table 1. Fine-tuning GPT-3.5 ranks best overall among our submissions and outperforms the baseline BART model in all aspects. Additionally, we experiment with different numbers of sentences being extracted by the TextRank (Table 2) and different numbers of training examples on the eLife validation set (Table 3). We observe no significant improvement over various evaluation metrics when increasing the number of TextRank sentences beyond 40 and the training set size beyond 200 examples.

Despite GPT-3.5's better performance over the smaller encoder-decoder models in most evaluation

[6] https://github.com/MaartenGr/KeyBERT

[7] https://platform.openai.com/docs/models/gpt-3-5-turbo

[8] https://openai.com/api/

# TextRank	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
20	0.4923	0.1498	0.4696	0.8647	9.1871	**7.4619**	10.6136	60.7699	0.6408	0.5460
30	0.5024	0.1525	0.4797	0.8647	9.1804	7.5366	10.5684	59.8206	0.6412	0.5352
40	0.5134	0.1566	0.4897	**0.8677**	**9.0398**	7.6474	**10.5426**	**61.9627**	0.6502	**0.5524**
50	0.5094	0.1563	0.4869	0.8661	9.1896	7.5420	10.5649	60.9104	0.6449	0.5466
100	**0.5151**	**0.1582**	**0.4907**	0.8667	9.5302	7.8078	10.8467	60.9124	**0.6563**	0.5432

Table 2: Ablation study of the number of sentences extracted by the TextRank for GPT-3.5 fine-tuning. The model is fine-tuned on 200 examples in each case, and evaluation is performed on the eLife validation set.

# Examples	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
w/o FT	0.3430	0.0786	0.3171	0.8360	15.6929	11.2094	17.4336	**65.3938**	**0.7467**	0.5106
100	0.49024	0.1449	0.4667	0.8625	9.9315	8.0081	11.2464	57.2724	0.6784	**0.5936**
200	**0.5134**	0.1566	**0.4897**	**0.8677**	**9.0398**	**7.6474**	**10.5426**	61.9627	0.6502	0.5524
400	0.5120	**0.1568**	0.4888	0.8673	9.3402	7.7173	10.7939	61.6123	0.6682	0.5580

Table 3: Ablation study of the number of training examples used to fine-tune GPT-3.5. All models use 40 sentences extracted by the TextRank. We apply the random seed as 42 for selecting examples. The w/o FT case uses a zero-shot prompting method to generate lay summaries. The prompt template for GPT-3.5 is provided in Appendix D.

Models	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
LED-base$_{4k}$	0.4724	0.1326	0.4503	0.8462	9.4983	**7.8822**	**10.0612**	67.4096	0.6282	**0.6093**
+ *DPR*	0.4822	0.1357	0.4592	0.8467	9.2784	7.9563	10.2329	68.4289	0.5760	0.5965
+ *Def*	0.4791	0.1347	0.4577	0.8466	9.2589	7.9413	10.2708	69.1213	0.5827	0.5981
+ *Ext*	0.4823	0.1347	0.4599	0.8466	**9.2203**	7.9210	10.2442	69.1202	0.6141	0.5902
+ *TR*	0.4818	0.1342	0.4601	0.8468	9.3755	7.9595	10.3095	68.5750	0.6129	0.5920
+ *All*	0.4810	0.1353	0.4582	0.8470	9.3195	7.9153	10.3041	68.6305	0.6150	0.5883
PubMed$_{LED4k}$	0.5070	0.1507	0.4770	0.8519	11.5237	8.9008	11.5916	69.7507	**0.6442**	0.5887
+ *All*$_{PubMed}$	**0.5140**	**0.1550**	**0.4868**	**0.8520**	10.3212	8.2847	10.6131	**70.7518**	0.6341	0.5883

Table 4: Ablation study of different model components on the eLife validation set. We use the same reordered sections of each article as base input. We apply PubMed LED large model for PubMed$_{LED4k}$ and All$_{PubMed}$ settings, and use LED-base model for all the other experiments due to limited computing resources. We report the result of PubMed$_{\text{LED}}$ model using *All* setting in Table 1.

aspects (Table 1), open-source models have some advantages, including reduced costs, the ability to fine-tune on various datasets, and reproducibility. Therefore, we conduct ablation studies on fine-tuning open-source models in Table 4. Initially, we compare two baseline configurations: the LED base model (LED-base$_{4k}$) and the PubMed LED large model (PubMed$_{LED4k}$), both using the top three sections as base input. We then apply the functional modules described in Section 2 to these baseline settings to assess the effectiveness of each component. In Table 4, *DPR* and *Def* refer to the RAG methods outlined in Section 2.3, which involve dense retrieval and entity-based definition retrieval from Wikipedia, respectively. *Ext* and *TR* denote the use of BERT-based unsupervised extractive summarization and TextRank, as introduced in Section 2.2.2. The term "*All*" represents the integration of all components, in the sequence of the top three sections, *Ext*, *DPR*, and *Def*, as input for fine-tuning. The results indicate the PubMed LED large model achieves the highest LENS and relevance scores when all components are included. However, the readability scores do not surpass those of the LED base model.

Meanwhile, we notice that an article's abstract achieves the highest factuality scores compared to the gold lay summary. Therefore, we explore the possibility of aligning the lay summaries generated by the PubMed LED model more closely with the article's abstracts through GPT-4 post-processing. However, our experimental results indicate no apparent improvement across most evaluation metrics when using GPT-4 to enhance the alignment of the generated lay summaries with the abstracts (experimental details in Appendix E).

4 Discussion and Conclusion

Applying the extract-then-summarize framework to fine-tune GPT-3.5 demonstrates superior performance in the biomedical lay summarization task compared to LED-based fine-tuning. The ablation

study indicates that incorporating external knowledge during model fine-tuning slightly enhances relevance metrics in the experiments of the LED-base model. However, it negatively impacts factuality scores, similar to the results observed when using extractive summarization and PubMed LED large model. None of the components enhance the factuality scores compared to the baseline settings, although there are improvements in relevance and readability scores (Table 4). We hypothesize that the external knowledge generated by RAG methods might contain noisy data, potentially affecting the factuality metrics. Additionally, the extractive summarizer may produce sentences with less contextual coherence than the original article, hindering the model's ability to understand causal information during fine-tuning. While increasing the model size enhances relevance scores, it decreases readability and factuality from the LED base model to the PubMed LED large model.

The case study detailed in Appendix F reveals that the GPT-3.5 and PubMed LED models produce unrelated information when creating lay summaries compared to the gold lay summary. Notably, GPT-3.5 produces longer summaries than the PubMed LED model despite both models having the same maximum decoding token limit of 512. Consequently, while GPT-3.5 includes more relevant sentences that closely match the original summary, it also introduces more irrelevant content.

Overall, fine-tuning GPT-3.5 with extractive summaries achieves the best overall ranking and highest relevance score in the BioLaySumm 2024 shared task, demonstrating the effectiveness of using key sentences from the article for LLM fine-tuning. The PubMed LED model, with additional functional components, also shows competitive results compared to GPT-3.5. Meanwhile, our findings using the PubMed LED model suggest a promising direction for future studies to develop advanced modules that combine extractive summarization and RAG to generate lay summaries, especially in improving the relevance scores and enhancing the accessibility of biomedical research.

Limitations

Our study's limitations are as follows: 1) We conduct experiments using LED-based models for only one epoch with a small batch size due to time and computational constraints. We hypothesize that the model's performance could vary with more tuning epochs. 2) Our section reordering method may miss sections that do not match the NLM dictionary of section names, potentially impacting the model's performance by omitting important content from articles. The proportions of mismatched sections are detailed in Appendix C. 3) The unsupervised extractive summarization methods used in this study are not tailored for lay summarization tasks, which may result in less relevant extraction. We suggest that developing a task-specific extractor could be a promising direction for future work. 4) We apply only two RAG methods in our experiments and concatenate the retrieved knowledge at the end of the input. The quality of the retrieved information was not filtered or verified, which may negatively impact fine-tuning performance. 5) Our method uses GPT-3.5 from OpenAI, which may not be fully reproducible since GPT-3.5 is a closed-source model and may update without unambiguous versions.

Acknowledgement

This work is partially supported by the National Library of Medicine (NLM) of the National Institutes of Health under the award number R01LM014292.

References

Ahsaas Bajaj, Pavitra Dangati, Kalpesh Krishna, Pradhiksha Ashok Kumar, Rheeya Uppaal, Bradford Windsor, Eliot Brenner, Dominic Dotterrer, Rajarshi Das, and Andrew Mccallum. 2021. Long document summarization in a low resource setting using pretrained language models. 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 71–80.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Sergey Brin and Lawrence Page. 1998. The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 30:107–117.

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)*.

Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283.

Edgar Dale and Jeanne S Chall. 1948. A formula for predicting readability: Instructions. *Educational research bulletin*, pages 37–54.

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 *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 468–477.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. 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.

Yue Guo, Wei Qiu, Gondy Leroy, Sheng Wang, and Trevor Cohen. 2024. Retrieval augmentation of large language models for lay language generation. *Journal of Biomedical Informatics*, 149:104580.

J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Huan Yee Koh, Jiaxin Ju, He Zhang, Ming Liu, and Shirui Pan. 2022. How far are we from robust long abstractive summarization? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2682–2698.

Philippe Laban, Tobias Schnabel, Paul N Bennett, and Marti A Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. 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.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020b. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

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.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A Learnable Evaluation Metric for Text Simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16383–16408.

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.

Derek Miller. 2019. Leveraging BERT for extractive text summarization on lectures. *arXiv preprint arXiv:1906.04165*.

OpenAI. 2024. gpt-3.5-turbo-1106. `https://platform.openai.com/docs/models/gpt-3-5-turbo`. Accessed: 2024-05-12.

Oisín Turbitt, Robert Bevan, and Mouhamad Aboshokor. 2023. MDC at BioLaySumm task 1: Evaluating GPT models for biomedical lay summarization. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 611–619, Toronto, Canada. Association for Computational Linguistics.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating Factual Consistency with A Unified Alignment Function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348.

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

A Experimental Settings

We apply two baseline model types in our experiments: LED base[9] (`allenai/led-base-16384`) and PubMed LED large[10] (`patrickvonplaten/led-large-16384-pubmed`) models. Our final submission uses GPT-3.5 as the base model[11].

Longformer Encoder-Decoder (LED). LED model is initialized from BART-base (Lewis et al.,

[9] `https://huggingface.co/allenai/led-base-16384`

[10] `https://huggingface.co/patrickvonplaten/led-large-16384-pubmed`

[11] `https://platform.openai.com/docs/models/gpt-3-5-turbo`

Models	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
LED$_{16k}$	0.4838	0.1378	0.4598	0.8475	9.5573	8.0395	10.2088	69.0407	0.6176	0.5837
LED$_{8k}$	0.4746	0.1342	0.4522	0.8463	9.5950	7.9085	10.2432	67.5068	0.6315	0.5903
LED$_{8k}$*	0.4750	0.1348	0.4530	0.8471	9.2274	7.9494	10.2126	69.8675	0.6391	0.6036
LED$_{4k}$†	0.4724	0.1326	0.4503	0.8462	9.4983	7.8822	10.0612	67.4096	0.6282	0.6093

Table 5: Performance comparison of various input lengths in the eLife dataset. All experiments are conducted under `led-base-16384`. 16k, 8k, and 4k are the maximum length of the model's input. * indicates that we restructure the input document in the order of abstract-background-conclusion-results-methods. † denotes we only input an article's abstract, background, and conclusion in model tuning.

Dataset	Section	Avg. Length (Train)	Avg. Length (Val)	Avg. Length (Test)
eLife	Article	13,942	13,705	11,683
	Lay Summary	437	445	-
PLOS	Article	8,963	8,925	9,039
	Lay Summary	239	239	-

Table 6: The average length of eLife and PLOS calculated by an average number of tokens. Article represents the full document of each article. Lay Summary is the gold summary of each article.

2020a) as both models share the same architecture, with a maximum input length of 16,384 tokens.

PubMed LED large. PubMed LED large model is fine-tuned on the PubMed Summarization dataset (Cohan et al., 2018) through the checkpoint of `led-large-16384`.

GPT-3.5. `GPT-3.5-turbo-1106` is fine-tuned on small random samples from the training datasets using the API provided by OpenAI.

We fine-tune the LED base and PubMed LED large models for 1 epoch and set batch size as 4. The maximum length of the decoder is 512. All experiments are conducted through one NVIDIA Tesla V100-32GB GPU. Considering memory efficiency, we use the default learning rate as 5e-5 for Adam optimization and set the floating point to 16 (i.e., fp16=True). For GPT-3.5 model fine-tuning[12], we apply the default API hyper-parameters, along with default values (i.e., epochs=3, batch size=1, learning rate multiplier=8).

We compare different input text lengths using the LED-base model in Table 5. The results indicate that decreasing the input length affects the relevance scores. Specifically, LED$_{16k}$, LED$_{8k}$, and LED$^{\dagger}_{4k}$ show a consistent decrease in relevance scores, while other evaluation metrics exhibit fluctuations. Notably, LED$_{16k}$ achieves the lowest factuality scores compared to other settings, suggesting a need to reduce the length of the input article to help the model capture more factual information. The encoder maximum length we use for both eLife and PLOS datasets in Table 1 is 8,192 tokens due to the computing limitation.

[12] https://platform.openai.com/docs/guides/fine-tuning

Dataset	# Train	# Validation	# Test
eLife	4,346	241	142
PLOS	24,773	1,376	142

Table 7: Statistics of LaySumm datasets.

Dataset	Split	# Positive Pairs	# Negative Pairs
eLife	Train	16,210	16,210
	Val	912	912
PLOS	Train	17,910	17,910
	Val	1,070	1,070

Table 8: Statistics of contrastive learning datasets. To balance the created datasets, we sample the same number of positive and negative pairs for each dataset.

Dataset	Train %	Val %
eLife	1.6 %	1.2 %
PLOS	0.6 %	0.4 %

Table 9: Unmatched section headings for eLife and PLOS datasets in section selection.

B Dataset Statistics

We report the average length of the article and lay summary in Table 6, as well as the statistics of two datasets in Table 7. As reported by Goldsack et al. (2022), there are no gold lay summaries for the test sets of both datasets for fair competition. In Table 8,

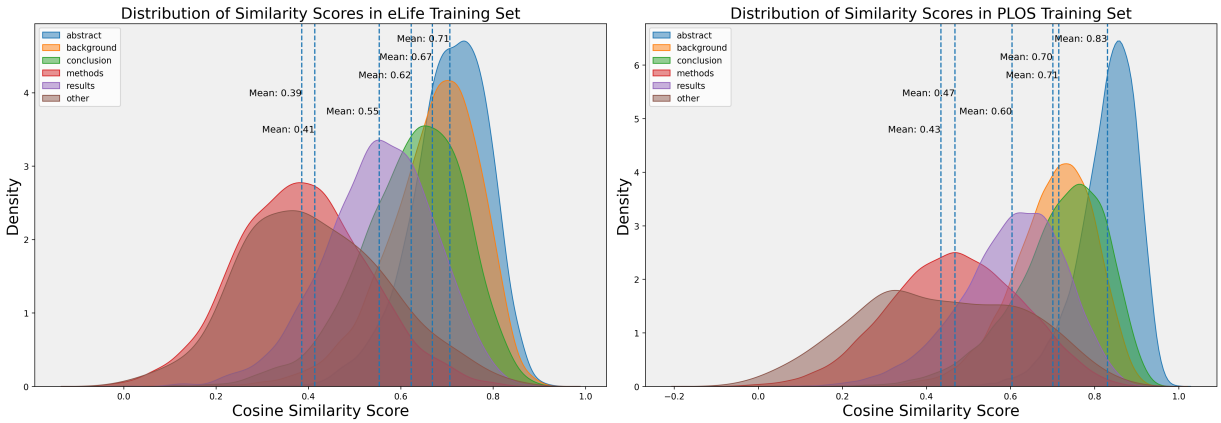

Figure 2: Comparison of section relevance in eLife and PLOS training sets grouped by NLM structured section labels. Density refers to the estimated probability density function of the cosine similarity scores for each section heading with the gold lay summary.

Models	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
LED$_{original}$	**0.4812**	**0.1355**	**0.4586**	0.8466	9.2523	**7.9069**	10.2308	67.9287	0.6224	0.5994
LED$_{ordered}$	0.4750	0.1348	0.4530	**0.8471**	**9.2274**	7.9494	**10.2126**	**69.8675**	**0.6391**	**0.6036**

Table 10: Evaluation of reordering sections in model tuning in the validation set of eLife. We set the input length of both models as 8192 tokens for equal comparison.

we show the contrastive learning datasets statistics for fine-tuning the embedding model introduced in Section 2.2.2.

C Section Relevance

As introduced in Section 2.2.1, we apply a section re-ranking strategy in our experiments to deal with long input lengths. We first pair the headings that appear in each dataset with the structured abstract section labels provided by NLM, which contains 3,032 section labels and 5 corresponding broader NLM categories: BACKGROUND, OBJECTIVE, METHODS, RESULTS, and CONCLUSIONS. We report the heading matching proportions in Table 9. Specifically, for the eLife dataset, we identify 339 unmatched headings in the training set and 14 in the validation set, out of 21,315 and 1,158 headings, respectively. The PLOS dataset has 833 unmatched headings in the training set and 28 in the validation set, with overall totals of 122,873 and 6,800 headings, respectively. Notably, no OBJECTIVE sections are matched in either the eLife or PLOS datasets. In these cases, we concatenate the unmatched sections to the end of the article. Subsequently, we rank the sections by calculating the cosine similarity between each section's content and the lay summary. We employ a pre-trained sentence transformer embedding model, `all-MiniLM-L6-v2`[13], to encode both the lay summary and each section, allowing us to compute similarity scores. Figure 2 depicts the section relevance distribution in eLife and PLOS training sets. Our findings indicate that the most relevant sections for both the eLife and PLOS datasets are the "Abstract", "Background", and "Conclusion", while the "Method" and mismatched "Other" sections are found to be less relevant.

Additionally, in Table 10, we compare the effectiveness of section reordering in an article by assessing the performance of models using ordered sections versus the original order. We use the LED-8k model on the eLife validation set for this evaluation. Specifically, we directly truncate the input as the natural order of the original dataset for the LED$_{original}$ model. Given our exploration of sections' relevance with gold lay summary (Figure 2), we re-rank the sections based on the order of abstract-background-conclusion-results-methods. Therefore, we reorder the input of the LED$_{re\text{-}order}$ model given the above order and truncate the model with an input limit of 8,192 tokens. Table 10 demonstrates that most evaluation scores improve with ordered sections as input, whereas ROUGE scores and DCRS show a decline.

[13] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

Models	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
Abs.	0.3189	0.0701	0.2934	0.8390	15.5000	11.7386	17.5873	38.3429	**0.9935**	**0.9488**
LaySum	**1.0000**	**1.0000**	**1.0000**	**1.0000**	10.8295	8.9364	12.4921	61.5753	0.5959	0.4943
P-LED$_L$	0.5140	0.1550	0.4868	0.8520	10.3212	8.2847	10.6131	70.7518	0.6341	0.5883
GPT-4	0.4979	0.1316	0.4668	0.8521	13.1813	9.9235	14.3912	70.3339	0.6669	0.5170

Table 11: Post-processing results in eLife validation set. Abs. and LaySum use the article's abstract and gold lay summary for evaluation. P-LED denotes PubMed LED large model (i.e., `patrickvonplaten/led-large-16384-pubmed`) and sets the input length as 8,192 tokens. GPT-4 represents the results using GPT-4 for post-processing.

D Prompt Template of GPT-3.5

We provide the prompt template for fine-tuning GPT-3.5 at the end of this section. The `### Article` represents the output of extractive summarization using TextRank introduced in Section 2.2.2. The `### Summary` denotes the gold lay summary of each article.

In the setting without fine-tuning (w/o FT), as introduced in Table 3, we use extractive summaries to prompt GPT-3.5 using the same template as fine-tuning GPT-3.5. The key difference in the w/o FT approach is that it does not include the gold-standard lay summary.

System:
Generate a lay summary of this biomedical article

User:
`### Article`:
Cell-fate reprogramming is at the heart of development...
`### Summary`:

E Post-Processing using GPT-4

We observe the abstracts of articles achieve higher factuality scores compared to the gold lay summaries. As illustrated in Table 11, abstracts attain the highest factuality scores, while gold lay summaries achieve full relevance scores. We assess whether an LLM can reassemble model-generated lay summaries to resemble the articles' abstracts more closely, thereby improving the factuality of the lay summaries. The input of GPT-4 includes a prompt template, original abstract, and PubMed LED model generated lay summary. While the AlignScore improves when using GPT-4 compared to the original settings, most other evaluation metrics, particularly readability scores, show a decline.

Due to the lack of factually accurate lay summaries as references to prompt or fine-tune GPT-4, the experiment is conducted solely under a zero-shot setting. We conclude that using zero-shot prompting with GPT-4 does not enhance the factuality of the generated lay summaries.

The prompt template used for GPT-4 is as follows:

System:
You will be provided with a biomedical abstract and a corresponding lay summary.

Your task is to enhance the lay summary by integrating factual information from the abstract. Consider the abstract as an additional reference.

Make sure to keep the same wording as the provided lay summary but add more factual information. Do not change any factual information. Do not reduce the readability and relevance of your enhanced lay summary. Do not make up information.

Keep your enhanced lay summary roughly the same length as the provided lay summary.

User:
`Abstract`: Cell-fate reprograming is at the heart of development...
`Lay Summary`: Genes are the building blocks of life...

F Case Study

To provide a more straightforward comparison of model-generated lay summaries, we conduct a case study comparing the generated lay summaries of GPT-3.5 and the PubMed LED large models in the

test sets of eLife and PLOS in Table 12 and Table 13. We randomly select an article with ID elife-78005-v1 from eLife and journal.pgen.1008255 from PLOS. Our findings reveal that both the fine-tuned GPT-3.5 and the PubMed LED model generate irrelevant information not mentioned in the gold lay summary in both datasets. However, GPT-3.5 produces more informative sentences than the PubMed LED large model.

Gold	When light enters the eyes , it is focused onto the retina , a thin layer of brain tissue at the back of the eye . The retina converts light information into electrical signals that are transmitted to the rest of the brain to perceive vision . Unlike the rest of the brain , this light-processing tissue can continue working even when removed from an animal , making it easier for scientists to study how the retina works . This has helped it become one of the best-understood parts of the brain . Most knowledge of retinal signal processing comes from studies of isolated retinas . However , it was still unclear if these samples behave the same way as they do in live animals , and whether findings in isolated retinas apply to natural visual processing in an awake state . To determine this , Boissonnet et al . compared the visual responses of the retina in awake mice , anesthetised mice and when isolated from mice . Measurements of retinal electrical signals showed that awake mice responded to light substantially more quickly and strongly than the others . Computational analysis suggested that the amount of information carried to the brain was largely comparable across the different subjects , but the retina in awake mice used more energy . The findings indicate that further studies are needed to better understand how the retina processes visual information in awake animals , rather than just in isolated conditions . Progressing this understanding could ultimately help to develop prosthetic devices that can act as a retina in the future .
GPT-3.5	The retina is a thin layer of tissue at the back of the eye that contains cells that are sensitive to light . These cells convert light into electrical signals that are sent to the brain , where they are processed to create the images that we see . The retina is one of the best understood parts of the brain , and many of the principles that govern how it works have been discovered by studying the retinas of animals in the lab . However , it is not clear how much of this knowledge applies to the retina in a living animal . This is because most of the studies have been done on retinas that have been removed from the eye and kept alive in a dish , or on retinas of animals that have been anesthetized . To investigate how the retina works in a living animal , Li et al . recorded the electrical activity of the retinas of mice while the animals were awake and while they were anesthetized . The experiments showed that the retinas of awake mice were more active than those of anesthetized mice . They also responded more quickly to changes in light and were able to detect a wider range of light levels . However , the retinas of awake mice were less efficient at transmitting information to the brain . This is because the increased activity in the retina did not result in a proportional increase in the amount of information that was sent to the brain . The findings of Li et al . suggest that the retina may use a different strategy to process information in living animals than it does in the lab . This is important because many of the principles that have been discovered by studying the retina in the lab are used to develop artificial retinas for people with vision loss . The findings of Li et al . suggest that these artificial retinas may need to be designed to work differently in living animals than they do in the lab .
PubMed$_{LED}$	The retina is the part of the brain that receives information from the eye and processes it into a visual image. The retina is made up of many different types of cells, but the activity of the retinal nerve cells is the most studied. The retinas are the most sensitive to light and are responsible for detecting color, contrast and motion. The activity of these cells is then processed by the brain, which then sends the information to the retina. However, it is not clear what exactly the retina tells the brain in awake animals. To investigate this question, Vlasiuk et al. recorded the electrical activity of retinal cells in the eyes of mice while they were awake, anesthetized and under anesthesia. The experiments show that the retinas of awake mice show faster firing rates and a larger range of responses than those of anesthetics-treated mice. This suggests that the retina uses a dense code of information in the retina, rather than a sparse code as had been assumed. The results also suggest that the brain uses dense coding in vivo. The next challenge will be to understand how the retina works in the brain and how it can be used to help us to understand the brain's role in visual processing. This could help us understand how we perceive the world around us, and how the brain responds to the world we see.

Table 12: A case study of article ID elife-78005-v1 generated by GPT-3.5 and PubMed LED large model on the eLife test set. Information highlighted in green indicates details mentioned in the gold lay summary, while red indicates details not included in the gold lay summary.

Gold	Gene inactivation in mice can be used to identify genes that are involved in important biological processes and that may contribute to disease . We used this approach to study the Ric-8B gene , which is highly conserved in mammals , including humans . We found that Ric-8B is essential for embryogenesis and for the proper development of the nervous system . Ric-8B mutant mouse embryos are smaller than their wild type littermates and show neural tube defects at the cranial region . This approach also allowed us to identify the biological pathways that potentially contribute to the observed phenotypes , and uncover a novel role for Ric-8B in the mTORC2 signaling pathway . mTORC2 plays particular important roles in the adult brain , and has been implicated in neurological disorders . Our mutant mice provide a model to study the complex molecular and cellular processes underlying the interplay between Ric-8B and mTORC2 in neuronal function .
GPT-3.5	mTOR is a serine/threonine protein kinase that is involved in a series of critical cellular processes . mTOR can be found in two functionally distinct complexes , mTORC1 and mTORC2 . In contrast to mTORC1 , little is known about the mechanisms that regulate mTORC2 . Here we show that mTORC2 activity is reduced in mice with a hypomorphic mutation of the Ric-8B gene . Ric-8B is a highly conserved protein that acts as a non-canonical guanine nucleotide exchange factor (GEF) for heterotrimeric Gs/olf type subunits . We found that Ric-8B hypomorph embryos are smaller than their wild type littermates , fail to close the neural tube in the cephalic region and die during mid-embryogenesis . Comparative transcriptome analysis revealed that signaling pathways involving GPCRs and G proteins are dysregulated in the Ric-8B mutant embryos . Interestingly , this analysis also revealed an unexpected impairment of the mTOR signaling pathway . Phosphorylation of Akt at Ser473 is downregulated in the Ric-8B mutant embryos , indicating a decreased activity of mTORC2 . Knockdown of the endogenous Ric-8B gene in cultured cell lines leads to reduced phosphorylation levels of Akt (Ser473) , further supporting the involvement of Ric-8B in mTORC2 activity . Our results reveal a crucial role for Ric-8B in development and provide novel insights into the signals that regulate mTORC2 .
PubMed$_{LED}$	mTOR, a serine/threonine protein kinase that is involved in a series of critical cellular processes, can be found in two functionally distinct complexes, mTORC1 and mTOR2. In contrast to mTOR1, little is known about the mechanisms that regulate mTORc2. Here we show that the Ric-8B gene is required for proper growth and nervous system formation during mouse embryogenesis. We found that the mutant embryos fail to close the neural tube at the cephalic region and die during mid-embryogenesis. Our results reveal a crucial role for Ric- 8B in development and provide novel insights into the signals that regulate the mTOR signaling pathway.

Table 13: A case study of article ID journal.pgen.1008255 generated by GPT-3.5 and PubMed LED large model on the eLife test set. Information highlighted in green indicates details mentioned in the gold lay summary, while red indicates details not included in the gold lay summary.