

Curriculum-Based Fine-Tuning for Summarization of Endometriosis Data

Stanford CS224N Custom Project

Ali Tout

Department of Computer Science
Stanford University
atout@stanford.edu

Abstract

Fine-tuned LLMs have become increasingly useful for quickly summarizing medical data that would otherwise require medical professionals to review large amounts of documentation over extended periods of time. However, a common issue with these models is hallucination, where the model may fabricate information or miss key clinical details within the data, particularly in areas such as planning, assessment, and symptoms [1].

In this work, I propose going beyond standard fine-tuning by implementing a three-stage curriculum learning strategy using QLoRA on the 0.6B parameter Qwen model. In this approach, the model is trained to sequentially learn key terms, main bullet points, and finally a full summary, allowing it to better understand which information within the clinical note is most relevant for summarization.

Results show that while the full curriculum learning pipeline does not outperform standard summary-only fine-tuning across ROUGE or BERTScore metrics, incorporating structured intermediate signals can still be beneficial. In particular, fine-tuning on key terms together with summaries achieves roughly a 33% improvement in BLEURT score compared to the non-fine-tuned baseline, suggesting that concise clinical guidance helps the model better capture the overall meaning of a medical note.

1 Key Information to include

- Mentor: Caroline Choi
- External Collaborators (if you have any): N/A
- Sharing project: N/A
- Late Days (if you are using any): 2

Team member	Late days remaining
<Ali Tout>	3
Total pooled	3
Team size	1
Late days applied for the report	2

2 Introduction

With the increasing use of Large Language Models (LLMs) to summarize clinical data, it is important to ensure that these models are capable of identifying the necessary information while avoiding hallucinations about a patient, which can be dangerous if the output is trusted [2]. Along with this, there are also significant costs associated with using large models to summarize large amounts of

data. If a general model such as GPT is used, it typically cannot be fine-tuned on private medical data, which often leads to worse results compared to using a locally fine-tuned model trained directly on clinical notes [3]. However, running and training a local model can also be expensive, as it requires powerful GPUs and substantial computational resources.

Being able to create a model that can accurately summarize medical data while not being prohibitively expensive would make it easier to identify patterns within Electronic Health Record (EHR) data, which could potentially lead to the discovery of new methods for treating disease. It would also allow healthcare professionals to understand complex patient information more quickly and at a deeper level.

To address these challenges, I implement a curriculum learning strategy where the model learns what information it should focus on when generating a summary. The model first learns to extract key terms from the clinical notes and generate bullet points describing the main ideas. It is then trained using the notes together with the extracted key terms to generate the main points of the document. Finally, the model is trained using the notes, key terms, and bullet points together to generate a detailed summary that explains the clinical note while reducing hallucination.

Results show that while the curriculum learning strategy improves BLEURT scores by around 33% compared to standard summary-only fine-tuning, it performs similarly to regular fine-tuning when evaluated using ROUGE and BERTScore metrics.

3 Related Work

This work is primarily inspired by the TriSum method proposed by [4], which employs a curriculum learning strategy for document summarization by first extracting structured reasoning components from a document before generating a summary. Specifically, they extract *aspects*, which represent short phrases describing key topics in a document, and *triples*, which structure information into a subject (*s*), relation (*r*), and object (*o*), which are then used to guide the model when producing the final summary. My approach follows a similar staged structure but adapts it for clinical data, replacing aspects and triples with key terms, bullet points, and summaries as the intermediate reasoning steps. This structure is designed specifically for medical notes, where one of the main challenges is ensuring that important clinical terms and symptoms are not missed during summarization. By explicitly training the model to recognize key clinical terms and generate structured bullet points before producing the final summary, the goal is to help the model focus on the most important medical information within the note.

More broadly, there has been significant recent interest in improving summarization quality through fine-tuning strategies. [5] explore fine-tuning open-source LLMs such as Llama2 and Mistral for aspect-based summarization, finding that fine-tuning on domain-specific data leads to meaningful improvements in summary quality over zero-shot baselines. This is particularly relevant to my work, as the key terms I use during training serve a similar role to the aspect signals they use to guide the model’s focus. [6] further investigate the trade-offs between in-context learning and parameter-efficient fine-tuning for summarization, showing that combining both approaches outperforms either alone, especially when only a limited number of annotated samples are available, a challenge I also face given the relatively small size of my clinical dataset. Finally, [7] demonstrate that QLoRA fine-tuning of LLaMA 2 on hospital discharge summaries can produce compact, clinically accurate summaries while keeping memory requirements manageable — directly motivating my choice to use QLoRA on a similarly small clinical dataset. Their finding that parameter-efficient fine-tuning outperforms prompt engineering alone on medical text further supports the fine-tuning approach taken in this work.

4 Approach

4.1 Curriculum Learning

To fine-tune the Qwen 0.6B model, I first use GPT-4o [8], a powerful reasoning model created by OpenAI, to generate the key terms, bullet points, and summaries for each document. The key terms represent the most clinically important topics within each note, giving the model an explicit signal for what information it should prioritize. The bullet points then bridge the gap between these key

terms and the final summary, allowing the model to practice structuring the most relevant clinical information before being asked to produce a cohesive paragraph. This staged structure is designed to reduce hallucination by ensuring the model has already identified the critical details before attempting to summarize. The following prompt was used for GPT-4o:

You are a clinical documentation assistant.

Given a medical note written by a physician, complete the following tasks:

1. Extract clinically relevant key terms grouped under the following categories:
 - Diagnoses / Conditions
 - Symptoms
 - Medications
 - Procedures / Planned Procedures
 - Imaging / Test Findings
 - Relevant History (medical, surgical, family, social)
 - Other clinically important details
2. Using those key terms, generate concise bullet points summarizing the main
 - ↪ clinical information.
 - Each bullet must begin with "- "
 - Focus only on clinically relevant information
 - Do not add interpretations or information not explicitly stated
3. Generate a cohesive clinical summary paragraph based on the key terms and bullet
 - ↪ points.
 - Maintain an objective clinical tone
 - Do not add new information
 - Avoid redundancy

Output format:

=====

Key Terms:

[Organized list]

Bullet Points:

- Bullet 1
- Bullet 2
- Bullet 3
- ...

Generated Summary:

[Concise clinical summary paragraph]

=====

The curriculum approach uses LLMs to sequentially generate key terms, bullet points, and finally a summary of a clinical document D . First, the model generates key terms K from the document, which can be represented probabilistically as

$$P(K \mid D) \tag{1}$$

Using both the document and the generated keywords, the model then produces bullet points B describing the note:

$$P(B \mid D, K) \tag{2}$$

Finally, to generate the summary S , the model conditions on the document, the keywords, and the bullet points:

$$P(S \mid D, K, B) \tag{3}$$

Following this staged approach, I first fine-tune Qwen to generate key terms given the document with the loss

$$L_K = - \sum_{D \in \mathcal{D}} \log p(K \mid D; \theta_s)$$

Next, I fine-tune the model to generate bullet points given the document and keywords, and then to generate summaries conditioned on the document, keywords, and bullet points with the following losses:

$$L_T = - \sum_{D \in \mathcal{D}} \log p(B \mid D, K; \theta_s)$$

$$L_S = - \sum_{D \in \mathcal{D}} \log p(S \mid D, K, B; \theta_s)$$

Finally, the fully trained model generates key terms, bullet points, and a summary using only the document as input. This process can be interpreted as optimizing the combined objective formed by the three losses above. All code used for training and experimentation was implemented by me (aside from standard libraries), as the original TriSum paper [4] did not provide publicly available code.

4.2 Model Architecture

QLoRA (quantized low-rank adaptation) was used for parameter-efficient fine-tuning of a Qwen3-0.6B backbone [9]. Qwen3-0.6B is a 600M parameter decoder-only transformer [10]. The model contains 28 transformer layers and uses grouped-query attention to improve inference efficiency. The base model weights were quantized to 4-bit NormalFloat in order to reduce the computational footprint while maintaining strong language modeling performance.

LoRA parameters were configured to balance model expressivity and computational efficiency. The rank (r) was set to 32 and the scaling factor (α) was set to 64, resulting in an α/r ratio of 2.0. A dropout rate of 0.05 was applied to the LoRA layers to help mitigate overfitting. The LoRA adapters were applied to all major linear projections within the transformer blocks, including the query, key, value, and output projections of the attention layers, as well as the gate, up, and down projections of the MLP layers. The Unsloth library, together with gradient checkpointing, was used to accelerate training and improve VRAM utilization.

4.3 Baselines

To evaluate the impact of different fine-tuning strategies and data structures, GPT-4o summaries were used as reference summaries when comparing six different model configurations:

- Base Qwen_notFinetuned: The original Qwen model utilized in a zero-shot capacity without further optimization
- Qwen FT, Summary Only: A fine-tuned variant optimized strictly based on existing summaries
- Qwen FT, Terms + Summary: Fine-tuned on both key terms and summaries (part of curriculum learning)
- Qwen FT, Terms + BP + Summary: Fine-tuned on key terms, bullet points, and summaries (part of curriculum learning)
- Qwen FT, Curriculum (Summary Extracted): The model predicts the full hierarchical; key terms, bullets, and summary structure, from which the summary is extracted
- Base Qwen (Thinking Mode): The base model uses a “thinking” (Chain-of-Thought) inference strategy without additional fine-tuning

These configurations allow us to evaluate how different levels of structured supervision influence the final summarization quality.

5 Experiments

5.1 Data

The FDA defines Real-World Evidence (RWE) as “clinical evidence about the usage and potential benefits or risks of a medical product derived from analysis of real-world data (RWD), which refers to data relating to patient health status and the delivery of health care routinely collected from a variety of sources.” RWD examples include case reports, retrospective observational studies, electronic health records, patient-generated data, wearable sensors, and measures of social determinants of health [11].

The University of California San Francisco (UCSF) Health and the San Francisco Department of Public Health (SFDPH) provide de-identified clinical data through Information Commons, which is a comprehensive ecosystem of data, tools, secure computation, and analysis environments. In this project, I use EHR clinical notes from UCSF for women with endometriosis between the ages of 15 and 50 who may or may not be taking hormonal contraceptives. These notes primarily document the evaluation and management of complex gynecologic conditions including endometriosis, chronic pelvic pain, infertility, pre-operative surgical planning, and post-operative follow-up.

In total, the dataset contains 1050 patient notes, with 900 used for training, 100 for evaluation, and 50 for testing. Each patient record includes an H&P (History and Physical), which is the most comprehensive assessment produced by physicians and is derived from patient interviews, physical examinations, and summaries of completed or pending tests [12].

5.2 Evaluation Method

The six model configurations described in Section 4.3 were evaluated by comparing their generated summaries against GPT-4o reference summaries. The goal is to determine whether the proposed curriculum learning strategy helps the model better understand the clinical notes and generate higher-quality summaries compared to simpler fine-tuning strategies.

Evaluation metrics included ROUGE-1, ROUGE-2, and ROUGE-L, which measure unigram overlap, bigram overlap, and the longest common subsequence between generated and reference summaries [13]. In addition, I report BERTScore F1 using BioClinicalBERT embeddings (embeddings trained for medical data), METEOR, and BLEURT. BERTScore measures semantic similarity using contextual embeddings [14], METEOR evaluates alignment using lexical and synonym matches with fragmentation penalties [15], and BLEURT is a learned metric designed to approximate human judgments of text generation quality [16].

5.3 Experimental Details

For training, I used the AdamW optimizer with a learning rate of 8×10^{-5} and a cosine learning rate scheduler to help reduce overfitting on the 900 training samples. Each phase of training was run for 4 epochs using a batch size of 8, with 100 samples reserved for evaluation.

For the second and third phases of training, the first two epochs use inputs consisting of the clinical note together with GPT-4o outputs. This provides higher-quality intermediate signals to help guide the model during the early stages of training, when the model’s own generations are still too noisy to be useful as training targets. However, the final two epochs instead use outputs generated by Qwen itself. This is done to reduce exposure bias, since in practice the model will not have access to GPT-generated intermediate outputs during inference. Without this step, the model would only ever learn to condition on high-quality GPT outputs, meaning it would likely struggle at test time when it must instead rely on its own generated key terms and bullet points, which are inevitably noisier. By gradually transitioning to self-generated inputs, the model learns to be robust to its own imperfections, better reflecting the conditions it will face during actual inference.

During generation, a temperature of 0.6 was used to allow some variability in the generated summaries while still maintaining stable outputs. To prevent repetitive text generation, the maximum repeated n -gram size was limited to 8, as I observed that the model would otherwise occasionally repeat the same bullet point multiple times.

5.4 Results

Table 1 shows the evaluation results for the six model configurations across ROUGE, BERTScore, METEOR, and BLEURT. Overall, fine-tuning on summaries alone achieves the strongest performance across most metrics, while incorporating additional structured supervision produces mixed results.

Model	R-1	R-2	R-L	R-Lsum	BERT-F1	METEOR	BLEURT
Base Qwen_notFinetuned	0.3678	0.1115	0.1971	0.1995	0.7605	0.2613	-0.4847
FT: Summary Only	0.5078	0.1905	0.2840	0.2842	0.8175	0.3379	-0.2439
FT: Terms + Summary	0.4836	0.1588	0.2523	0.2522	0.8151	0.3250	-0.1628
FT: Terms + BP + Summary	0.4878	0.1594	0.2574	0.2574	0.8132	0.3290	-0.1671
FT: Curriculum (Summary Extracted)	0.4371	0.1384	0.2361	0.2366	0.7933	0.2870	-0.4042
Base Qwen (Thinking Mode)	0.4093	0.1189	0.2168	0.2177	0.7783	0.2532	-0.4311

Table 1: Evaluation results for the six Qwen model configurations across ROUGE, BERTScore (F1), METEOR, and BLEURT metrics.

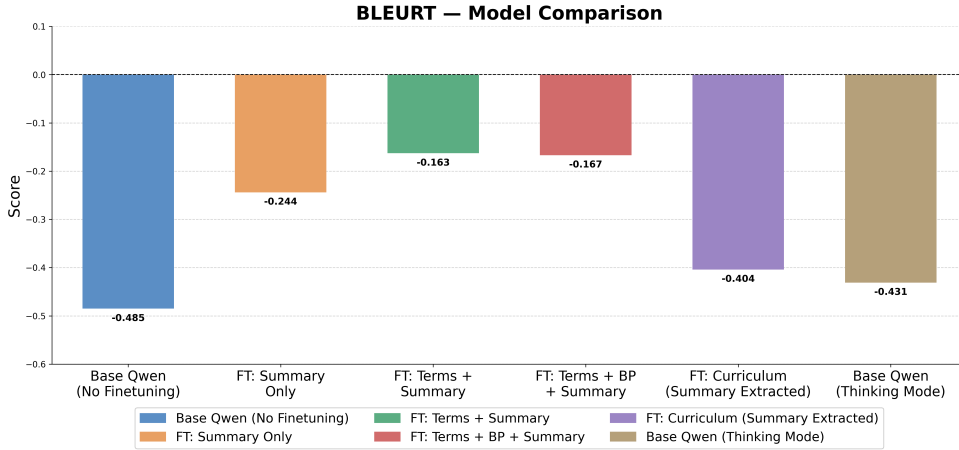


Figure 1: BLEURT scores for the six Qwen configurations.

This can also be compared to the results achieved in the original paper (Jiang et al., 2024):

Model	CNN/DailyMail				XSum				ClinicalTrial			
	R-1	R-2	R-L	Δ	R-1	R-2	R-L	Δ	R-1	R-2	R-L	Δ
Baselines												
BERTSumAbs	41.2	18.7	37.2	+13.6%	38.8	16.5	31.0	+28.3%	39.2	19.3	29.6	+19.3%
T5Large	42.4	20.8	39.9	+7.0%	40.1	17.2	32.3	+23.5%	41.3	22.1	32.5	+9.6%
BARTLarge	44.0	21.1	40.6	+4.4%	45.4	22.3	37.3	+5.4%	43.5	23.3	33.7	+4.6%
PEGASUS	44.2	21.6	41.3	+3.0%	46.7	24.4	38.9	+0.6%	41.8	22.9	31.7	+9.0%
GSum	45.5	22.3	42.1	+0.4%	45.1	21.5	36.6	+7.3%	43.5	23.1	32.8	+5.7%
BigBirdLarge	43.8	21.1	40.7	+4.5%	47.1	24.1	38.8	+0.6%	44.2	23.8	34.5	+2.5%
SimCLS	45.6	21.9	41.0	+1.7%	46.6	24.2	39.1	+0.7%	43.8	23.3	34.1	+3.9%
SeqCo	45.0	21.8	41.8	+1.6%	45.6	22.4	37.0	+5.4%	42.8	22.5	33.2	+6.7%
GLMRoBERTa	43.8	21.0	40.5	+4.7%	45.5	23.5	37.3	+4.1%	43.3	23.0	33.9	+4.9%
GPT-3.5 zero-shot	37.4	13.8	29.1	+37.4%	26.6	6.7	18.8	+112.5%	34.8	12.8	23.5	+47.8%
Our Method												
GPT-3.5 w/ TriSum rationale	46.7	23.5	40.7	-0.5%	34.4	12.6	28.4	+46.8%	44.6	24.5	30.4	+5.6%
TriSum-S	45.9	22.8	42.3	-0.6%	47.4	24.8	39.4	-1.0%	45.3	24.8	35.0	+0.0%
TriSum-C	45.5	22.3	41.2	+1.2%	46.5	24.0	38.7	+1.1%	44.2	23.7	34.4	+2.7%
TriSum-J	45.7	22.7	41.9	—	47.3	24.4	39.0	—	45.3	24.6	35.2	—

Table 2: ROUGE evaluation results across CNN/DailyMail, XSum, and ClinicalTrial datasets. ...

The most important metric was BLEURT as it is trained off human evaluations of summaries (albeit not specifically for medical data). As shown in Figure 1, using parts of the Curriculum Learning method allows the model to attain the highest BLEURT score; however, full Curriculum Learning does not perform well.

In the original paper, we can clearly see improvements in ROUGE through the curriculum learning method they use, which is not the case with my model. It is worth noting that these results are not directly comparable, as TriSum was evaluated on CNN/DailyMail, XSum, and ClinicalTrial datasets, while my model was evaluated on a much smaller private dataset of endometriosis clinical notes.

6 Analysis

6.1 Limitations

From the results, we can see that the Curriculum Learning strategy does not perform significantly better than regular summary fine-tuning. In fact, because the model is required to generate longer outputs consisting of key terms, bullet points, and the final summary, this sometimes leads to hallucinations and ultimately produces summaries that are worse than those generated by the summary-only fine-tuned model. Fine-tuned models also appear to prioritize using more medical terminology, even when the reference summaries generated by GPT do not contain as many clinical terms. Figure 2 further illustrates how curriculum-based models tend to generate more medical terminology than the reference summaries.

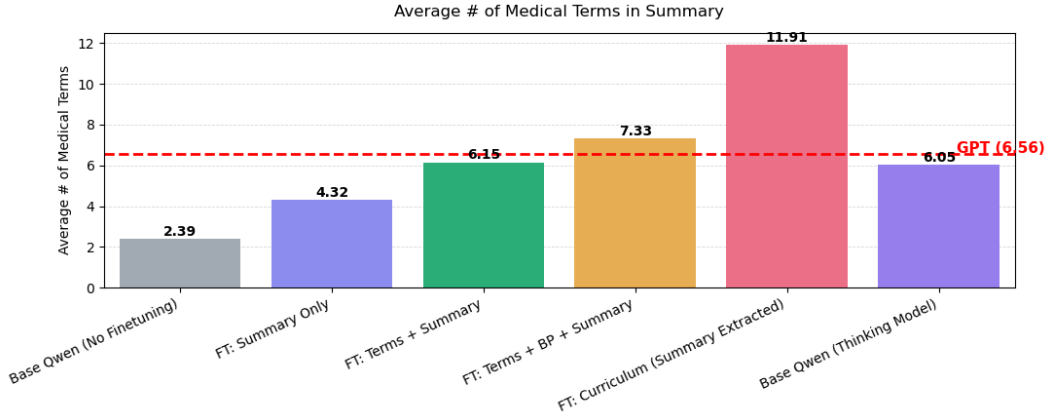


Figure 2: Average number of medical terms used in generated summaries across the six model configurations.

However, we also observe that having the model first generate key terms and/or bullet points and then feeding them back into the model to produce the final summary can still be beneficial. In particular, the model achieves approximately a 33% improvement in BLEURT score under this setup, although the other evaluation metrics remain roughly the same.

One of the main limitations of this work is the size of the dataset. The model was trained on only 900 clinical notes, which is likely not sufficient for the model to fully learn patterns within gynecology-related clinical data. This relatively small dataset also increases the risk of overfitting. As a result, I had to use a relatively small learning rate ($8e-5$) to prevent the Qwen3 0.6B model from overfitting too quickly. Although I implemented evaluation callbacks to monitor performance and ensure the model did not fully overfit, the model still showed a strong tendency toward overfitting, highlighting the limitations imposed by both the dataset size and model capacity.

Another limitation relates to the reproducibility of the TriSum method used as inspiration for this work. In the original paper [4], the authors report that their TriSum model performs better when using curriculum learning. They describe their approach as transferring summarization ability from a very large language model ($\geq 100B$ parameters) to a smaller model ($\leq 1B$ parameters) through rationale probing and curriculum learning. However, the model used in their experiments is GPT-3.5 Turbo,

which is estimated to contain at least 20 billion parameters according to Microsoft, and the paper does not clearly demonstrate experiments using models below the 1B parameter range.

Additionally, while the paper provides a detailed conceptual description of their curriculum learning framework, it does not include any publicly available implementation or detailed training setup. As a result, much of my implementation required experimentation and trial-and-error when selecting training procedures and hyperparameters. These differences in implementation may have contributed to some of the limitations observed in my results.

6.2 Future Work

From the results, we can see that the most impactful improvement beyond simple summary fine-tuning comes from incorporating key terms during training. However, the key terms used in this work may be too verbose for a smaller model such as Qwen 0.6B to handle effectively. In future work, it would likely be beneficial to use more concise key terms, potentially restricting each category to only a few important words.

Another possible direction would be to test the proposed curriculum learning strategy on a larger model. A model with more parameters may be better able to handle longer structured outputs and generate summaries with fewer hallucinations. Exploring this approach could help determine whether the limitations observed in this work are primarily due to model capacity or the curriculum learning strategy itself.

7 Conclusion

In this project, I explored whether a curriculum learning strategy could improve the quality of clinical note summarization using a relatively small language model. Specifically, I implemented a three-stage training pipeline in which the model learns to generate key terms, bullet points, and finally a summary of the clinical document. The goal of this approach was to encourage the model to focus on the most important clinical information before generating the final summary.

The experimental results show that while the full curriculum learning pipeline does not outperform standard summary fine-tuning across most evaluation metrics, incorporating structured intermediate signals can still be beneficial. In particular, fine-tuning on key terms together with summaries produces the highest BLEURT score, suggesting that explicitly guiding the model toward important clinical concepts helps it better capture the overall meaning of the medical note. However, extending this structure to include longer outputs such as full bullet point rationales introduces additional complexity and appears to increase hallucination, especially for a smaller model such as Qwen 0.6B.

Overall, these findings suggest that structured supervision can improve semantic quality in clinical summarization, but the amount of intermediate structure must be carefully balanced with model capacity and dataset size. While the full curriculum learning strategy did not outperform simpler approaches in this work, the results indicate that incorporating concise clinical signals such as key terms may still be a promising direction for improving summarization models trained on limited medical datasets.

Future work could explore applying this curriculum framework to larger language models or using more compact intermediate representations that are easier for smaller models to learn. Such approaches may help better leverage structured reasoning signals while reducing hallucinations in clinical summarization tasks.

References

- [1] Elham Asgari, Nina Montana Brown, Magda Dubois, Saleh Khalil, Jasmine Balloch, Joshua Yeung, and Dominic Pimenta. A framework to assess clinical safety and hallucination rates of llms for medical text summarisation. *npj Digital Medicine*, 8, 05 2025.
- [2] Rami Hatem, Brianna Simmons, and Joseph E Thornton. A call to address ai “hallucinations” and how healthcare professionals can mitigate their risks. *Cureus*, 15(9), 2023.

- [3] Michael L. Burns, Ssu-Ying Chen, Chu-An Tsai, John Vandervest, Balaji Pandian, Paige Nong, David A. Hanauer, Andrew Rosenberg, and Jodyn Platt. Generative ai costs in large healthcare systems, an example in revenue cycle. *npj Digital Medicine*, 8(1):579, 2025.
- [4] Pengcheng Jiang, Cao Xiao, Zifeng Wang, Parminder Bhatia, Jimeng Sun, and Jiawei Han. TriSum: Learning summarization ability from large language models with structured rationale. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2805–2819, Mexico City, Mexico, June 2024. Association for Computational Linguistics.
- [5] Ankan Mullick, Sombit Bose, Rounak Saha, Ayan Kumar Bhowmick, Aditya Vempaty, Pawan Goyal, Niloy Ganguly, Prasenjit Dey, and Ravi Kokku. Leveraging the power of llms: A fine-tuning approach for high-quality aspect-based summarization, 2024.
- [6] Che Guan, Andrew Chin, and Puya Vahabi. Enhancing news summarization with elearnfit through efficient in-context learning and efficient fine-tuning, 2024.
- [7] Joyeeta Goswami, Kaushal Kumar Prajapati, Ashim Saha, and Apu Kumar Saha. Parameter-efficient fine-tuning large language model approach for hospital discharge paper summarization. *Applied Soft Computing*, 157:111531, 2024.
- [8] OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoochian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codisputi, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogu Gierler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O’Connell, Ian O’Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang,

Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljube, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunningham, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. Gpt-4o system card, 2024.

- [9] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms, 2023.
- [10] Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models, 2025.
- [11] Lakshmi Radhakrishnan, Gundolf Schenk, Kathleen Muenzen, Boris Oskotsky, Habibeh Ashouri Choshali, Thomas Plunkett, Sharat Israni, and Atul J Butte. A certified de-identification system for all clinical text documents for information extraction at scale. *JAMIA Open*, 6(3):ooad045, 07 2023.
- [12] J. Lucian Davis and John F. Murray. History and physical examination. In V. Courtney Broadus, Robert J. Mason, Joel D. Ernst, Talmadge E. King Jr., Stephen C. Lazarus, John F. Murray, Jay A. Nadel, Arthur S. Slutsky, and Michael B. Gotway, editors, *Murray and Nadel’s Textbook of Respiratory Medicine*, pages 263–277.e2. Elsevier, 2016.
- [13] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.
- [14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.
- [15] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Jade Goldstein, Alon Lavie, Chin-Yew Lin, and Clare Voss, editors, *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.

- [16] Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. Bleurt: Learning robust metrics for text generation, 2020.