YaleNLP @ PerAnsSumm 2025: Multi-Perspective Integration via Mixture-of-Agents for Enhanced Healthcare QA Summarization

Dongsuk Jang[1,2,3], Alan Li[1], Arman Cohan[1]
[1]Department of Computer Science, Yale University,
[2]Interdisciplinary Program for Bioengineering, Seoul National University,
[3]Integrated Major in Innovative Medical Science, Seoul National University
{james.jang, haoxin.li, arman.cohan}@yale.edu

Abstract

Automated summarization of healthcare community question-answering forums is challenging due to diverse perspectives presented across multiple user responses to each question. The PerAnsSumm Shared Task was therefore proposed to tackle this challenge by identifying perspectives from different answers and then generating a comprehensive answer to the question. In this study, we address the PerAnsSumm Shared Task using two complementary paradigms: (i) a training-based approach through QLoRA fine-tuning of LLaMA-3.3-70B-Instruct, and (ii) agentic approaches including zero- and few-shot prompting with frontier LLMs (LLaMA-3.3-70B-Instruct and GPT-4o) and a Mixture-of-Agents (MoA) framework that leverages a diverse set of LLMs by combining outputs from multi-layer feedback aggregation. For perspective span identification/classification, GPT-4o zero-shot achieves an overall score of 0.57, substantially outperforming the 0.40 score of the LLaMA baseline. With a 2-layer MoA configuration, we were able to improve LLaMA performance up by 28% to 0.51. For perspective-based summarization, GPT-4o zero-shot attains an overall score of 0.42 compared to 0.28 for the best LLaMA zero-shot, and our 2-layer MoA approach boosts LLaMA performance by 32% to 0.37. Furthermore, in few-shot setting, our results show that the sentence-transformer embedding-based exemplar selection provides more gain than manually selected exemplars on LLaMA models, although the few-shot prompting is not always helpful for GPT-4o. The YaleNLP team's approach ranked the overall second place in the shared task.

1 Introduction

Healthcare Community Question Answering (CQA) forums are rapidly growing as accessible platforms for individuals to seek medical advice, share personal experiences, or request simplified explanations of health conditions. Unlike expert-oriented medical sites, user-driven forums incorporate a broad range of viewpoints, from anecdotal evidence to speculative reasoning. Although the diversity can enrich the discussion, it also leads to information overload and frequent off-topic comments, making it difficult for newcomers to identify critical insights. Traditionally, the CQA answer summarization task focuses on a single best-voted answer (Chowdhury and Chakraborty, 2018; Chowdhury et al., 2020) as a reference summary. However, a single answer often fails to capture the diverse perspectives presented across multiple answers. Providing the answers in structured, perspective-specific summaries could better serve the information needs of end users.

In response to this challenge, the **PerAnsSumm Shared Task** at the CL4Health@NAACL 2025 Workshop (Agarwal et al., 2025) introduces a perspective-specific summarization benchmark, encouraging researchers to design systems that explicitly recognize and integrate various user viewpoints into their outputs. The task is comprised of two phases. Given a medical related query and a set of answers from CQA forums, the system is required to (i) identify the specific perspective in each of the answer and (ii) generate a summarization for each of these perspectives across different answers. Detailed task setup will be introduced in Section 3.

Our main contributions and findings are as follows:

- We show that GPT-4o (OpenAI et al., 2024) generally outperforms 70B-level open-source models (the largest models we have access to) in both the span identification/classification and perspective-based summarization tasks. Providing few-shot examples do not *consistently* yield higher performance.
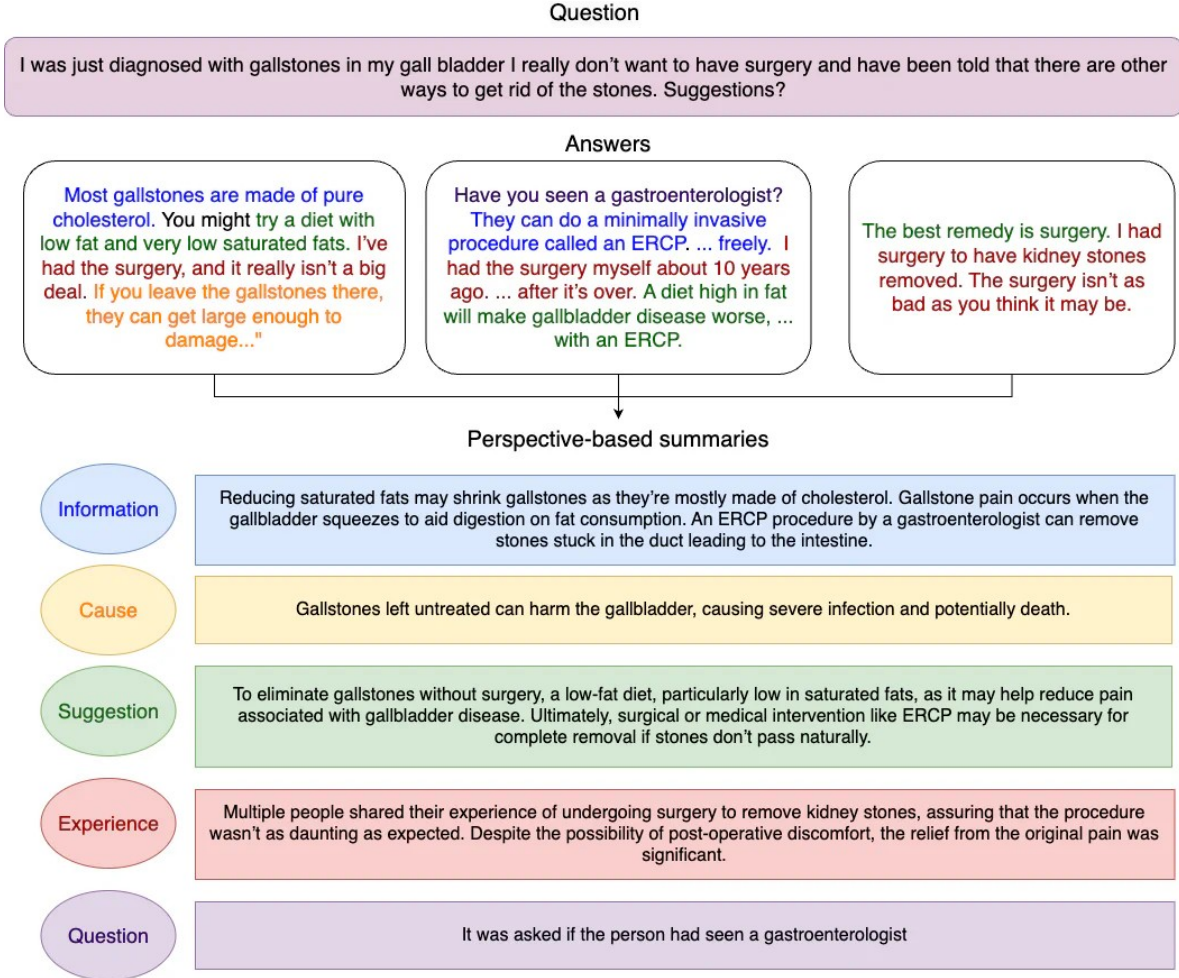
---

[1]Figure adapted from Agarwal et al. https://peranssumm.github.io/docs/

Figure 1: PerAnsSumm Shared Task overview.[1]

- In few-shot setting, example selection by clustering on candidate example embeddings yield consistent improvements over manual example selection.

- Implementing a MoA (Wang et al., 2024) approach with multiple open-source LLMs significantly improves performance over individual models, demonstrating the potential of this ensemble strategy.

- QLoRA (Dettmers et al., 2023) fine-tuning with generic limited training data does not provide performance gains under our experimental conditions; in fact, it degrades performance. Due to the time constraints of the challenge, we were unable to explore additional fine-tuning configurations. We leave fine-tuning recipe exploration to future work.

We reimplemented[2] the relevant techniques to align

with the PerAnsSumm Shared Task. Through these experiments, our objective is to provide insight into the strengths and limitations of LLM-based approaches for perspective-aware summarization in medical CQA.

2 Related Work

Early abstractive summarization largely relied on pre-trained models such as BART (Lewis et al., 2020), T5 (Raffel et al., 2023), and PEGASUS (Zhang et al., 2020a), demonstrating strong performance on news benchmarks like CNN/DailyMail (Hermann et al., 2015) or XSum (Narayan et al., 2018). Yet, these approaches are typically optimized for well-structured, professionally written content. In contrast, healthcare forums contain personal opinions, anecdotal evidence, and multiple viewpoints that can hinder purely data-driven summarizers (Chaturvedi et al., 2024).

Recent works in aspect- or perspective-oriented summarization highlight the value of parsing out

different user viewpoints. Naik et al. (2024) emphasize splitting content into categories like *cause, suggestion, experience*, while AnswerSumm (Fabbri et al., 2022) extracts sentence-level spans for query-focused summaries, though it does not fully address overlapping perspectives common in community Q&A. At the same time, multi-document techniques for high-variance domains (Liu et al., 2018) suggest strategies for aggregating and reconciling disparate user responses.

Moreover, the rise of large language models has fueled interest in zero-/few-shot prompting, with studies showing that manually curated exemplars can be fragile or insufficiently general. Embedding-based selection methods like FsPONER (Tang et al., 2024) and adaptive few-shot prompting (Tang et al., 2025; Chang et al., 2021) propose retrieving exemplars via similarity or clustering, offering more stable and domain-sensitive prompts. Such techniques are well-suited to healthcare Q&A, where a single misaligned exemplar can skew the summary toward incorrect or irrelevant details.

While prompt-based approaches can reduce reliance on large labeled datasets, certain tasks still benefit from specialized model tuning. To this end, LoRA (Hu et al., 2021) introduced a low-rank adaptation mechanism that updates only a small fraction of model parameters, and QLoRA (Dettmers et al., 2023) extends this concept by quantizing weights for further efficiency. These methods enable domain-focused tuning without the prohibitive resource costs typically associated with training massive LLMs from scratch.

3 PerAnsSumm

PerAnsSumm Shared Task is comprised of two sub-tasks sequentially, as shown in . Given a question Q, a set of answers A, and perspective categories {cause, suggestion, experience, information, question}, we are assigned the following two tasks:

3.1 Task A: Span Identification and Classification

For each answer in A, identify all text spans that convey any of the five perspectives.

Following the task guidelines, systems must output a list of labeled spans. For example:

```
span: "<extracted span>", label:
"<perspective>"
```

Any text not relevant to a predefined perspective is omitted.

Evaluation Metrics PerAnsSumm evaluates Task A under two main criteria:

- **Classification**: Whether the model correctly assigns a perspective label to an answer if it contains that perspective. Macro-F1 and Weighted-F1 are reported.

- **Span Matching**: Compares predicted spans with gold-standard spans via strict matching and proportional matching.

An overall macro-average of these measures is used for final ranking.

3.2 Task B: Perspective-Based Summaries

Building on Task A, after identifying and labeling spans in a Q&A thread, the system must produce a short, coherent summary for each perspective that appears.

Systems typically generate summaries in a structured format, for example:

```
EXPERIENCE Summary: <text>
INFORMATION Summary: <text>
CAUSE Summary: <text>
SUGGESTION Summary: <text>
QUESTION Summary: <text>
```

If a perspective is deemed absent by the model, no summary is produced for that label.

Evaluation Metrics To assess each perspective-specific summary, PerAnsSumm combines measures of *relevance* and *factuality*,

- **Relevance**: ROUGE (Lin, 2004), BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and BERTScore (Zhang et al., 2020b) quantify how well the generated summary aligns with the reference.

- **Factuality**: AlignScore (Zha et al., 2023) and SummaC (Laban et al., 2022) confirm that the summary is consistent with the original source text (i.e., it does not hallucinate or contradict).

These sub-metrics are aggregated into a final Task B score.

4 Methods

This section details the various modeling strategies we explore, including zero-/few-shot prompting, the MoA framework, and QLoRA supervised fine-tuning.

4.1 Zero-/Few-Shot Prompting

Zero-Shot Setup. We first experiment with prompting large language models using a instruction that specifies the task (either span identification/classification or perspective-based summarization). For instance, we provide definitions of the five perspectives (*cause, suggestion, experience, information, question*) and ask the model to extract or summarize accordingly (best prompt for each tasks are detailed in Appendix C). This approach requires no additional training or fine-tuning, leveraging the general knowledge embedded in instruction-tuned LLMs.

Few-Shot Setup. We provide 3–5 exemplars to the model via the prompt. We investigate two distinct methods for exemplar selection:

Manually Curated: We pick representative CQA threads that cover multiple perspectives and exhibit typical corner cases.

Embedding-Based Selection: We embed all potential demonstration samples from training set with a sentence-transformer (e.g., `all-MiniLM-L6-v2` in our case), cluster them using k-means, and then pick top-k samples based on proximity to the test query.

4.2 Mixture-of-Agents

Medical content requires both domain knowledge and nuanced understanding of different viewpoints. To overcome these limitations, we implement a Mixture-of-Agents (MoA) framework that leverages the complementary strengths of multiple language models working in concert. MoA is a framework for ensembling multiple sub-models (or agents) and integrating their outputs via an aggregator. We adapt and extend this method for our tasks. Specifically, we consider different numbers of layers (1, 2, or 3) in the MoA pipeline:

- **1-Layer MoA**: Each agent generates a partial response (e.g., predicted spans or short perspective-based summaries). An aggregator model then fuses these responses into a final output in a single step.

- **2-Layer MoA**: After collecting agent outputs, we employ an intermediate "verification" layer to refine or check consistency before passing the refined results to the final aggregator model.

- **3-Layer MoA**: We add an additional "hallucination detection" layer, which aims to filter out or correct unsupported statements before the final aggregation.

For our agent selection, we incorporate diverse models including open-source LLMs (LLaMA-3.3-70B-Instruct, Qwen-2.5-72B-Instruct, Deepseek-R1-Distill-LLaMA-70B) and closed-source models (GPT-4o, GPT-4o-mini). This diversity is intentional—each model brings different strengths in medical reasoning, language understanding, and factual recall. By combining them, we aim to create a system that outperforms any individual model, especially for complex medical content where perspectives might be subtle or require domain expertise.

We test various configurations to understand the optimal MoA architecture for each subtask. These configurations include combinations of open-source models only, GPT-4o only, and hybrid approaches where different model types handle different stages of the pipeline. For example, one effective arrangement uses GPT-4o for span identification/classification and a MoA ensemble for perspective-wise summarization based on those identified spans. We also explore the reverse configuration, as well as using MoA for both tasks. Through these experiments, we can measure the synergistic effects gained from mixing diverse LLMs and identify which models perform best at each stage of the process.

The multi-layer verification approach is particularly valuable for healthcare content, where accuracy is paramount. By adding verification and hallucination detection layers, we create checkpoints where potentially incorrect or unsupported information can be filtered or corrected before final aggregation, improving the reliability of the generated summaries.

4.3 QLoRA Supervised Fine-Tuning

While zero-/few-shot prompting relies on the generalization capabilities of LLMs, we also investigate QLoRA, a parameter-efficient fine-tuning approach. Through QLoRA, we can update a small set of low-rank adaptation parameters while keeping the majority of model weights frozen. This reduces both the computational overhead and memory usage compared to full fine-tuning.

Model	Setting	M-F1	W-F1	St-P	St-R	St-F1	Pr-P	Pr-R	Pr-F1	Overall
LLaMA-3.3-70B-Instruct	Zero-shot	0.5381	0.7299	0.0320	0.1218	0.0507	0.4530	0.6991	0.5498	0.3968
	3-shot w/ H	0.5390	0.7265	0.0339	0.1240	0.0513	0.4665	0.7163	0.5673	0.4031
	3-shot w/ C	0.5697	0.7676	0.0385	0.1311	0.0565	0.4954	0.7404	0.5974	0.4246
	QLoRA SFT	0.4788	0.6584	0.0256	0.1158	0.0447	0.4216	0.6681	0.5184	0.3664
GPT-4o	Zero-shot	**0.8949**	**0.9190**	**0.1756**	**0.2641**	**0.2110**	0.6578	0.7392	0.6961	**0.5697**
	3-shot w/ H	0.8176	0.8479	0.1552	0.2193	0.1818	0.6145	0.7124	0.6599	0.5261
	3-shot w/ C	0.8553	0.8723	0.1468	0.2546	0.1862	**0.6810**	**0.7525**	**0.7150**	0.5580
MoA	Best 1	0.8129	0.8478	0.1491	0.2072	0.1734	0.5512	0.6942	0.6145	0.5063
	Best 2	0.7682	0.7809	0.1443	0.1697	0.1560	0.5412	0.6512	0.5912	0.4753

Table 1: **Task A**(span identification/classification) results . "3-shot w/ H(uman)" means three manually curated examples were used for few-shot prompting; "3-shot w/ C(lustering)" means three exemplars were automatically selected via sentence-transformer embeddings. Metrics include Macro-F1 (M-F1), Weighted-F1 (W-F1), Strict Precision/Recall/F1 (St-P, St-R, St-F1), Proportional Precision/Recall/F1 (Pr-P, Pr-R, Pr-F1), and an Overall average.

5 Experiments

We employ a diverse set of open-source models (LLaMA-3.3-70B-Instruct (AI, 2024), Qwen-2.5-72B-Instruct (Qwen et al., 2025), and Deepseek-R1-Distill-LLaMA-70B (DeepSeek-AI et al., 2025)) and closed frontier model (GPT-4o and GPT-4o-mini (OpenAI et al., 2024)).

5.1 Experimental Data

We employ the PUMA[3] (Naik et al., 2024) corpus provided by PerAnsSumm shared task. It contains 3,245 Q&A threads, each with up to five perspective annotations (cause, suggestion, experience, information, question) and reference summaries per perspective. We follow official splits: 2,236 threads for training, 959 for validation, and 50 withheld for testing, while for the paper, we tested on the last 400 cases from valid set.

5.2 QLoRA Finetuning Implementation

We used llama-factory (Zheng et al., 2024) toolkit to simply fine-tune LLaMA-3.3-70B-Instruct under various hyperparameter settings. For additional fine-tuning details, see Appendix A.

6 Results

We evaluate our approaches on two tasks: **Task A** (span identification/classification) and **Task B** (perspective-based summarization), using the macro-averaged metrics described in Section 3.1 and 3.2.

---

[3]**P**erspective s**UM**marization d**A**taset

6.1 Task A: Span Identification and Classification

Table 1 presents the classification and span-matching results.

GPT-4o Zero-Shot remains the best overall single-model approach, scoring 0.5697 in Overall average, which notably outperforms all other models or methods. Detailed span identification results is described in Appendix B.

Few-Shot Prompting For both LLaMA-3.3-70B-Instruct and GPT-4o, embedding-based selection (0.4246 and 0.5580 overall) outperforms manually curated exemplars (0.4031 and 0.5261), showing better generalizability than human-chosen examples.

QLoRA Supervised Fine-tuning For Task A, our QLoRA-based fine-tuning of the LLaMA-3.3-70B-Instruct model (see Table 1) obtains an overall score of 0.3664, which is below the best zero- or few-shot baselines.

MoA Details. **Best 1** is a 2-layer MoA with four open-source models in Layer 1 (two LLaMA-3.3-70B-Instruct + two Qwen-2.5-72B-Instrcut), one LLaMA-3.3-70B-Instruct in Layer 2, and an aggregator also based on LLaMA-3.3-70B-Instruct. As illustrated in Figure 2, a 2-layer configuration strikes the best balance between thoroughness and retaining valid outputs, outperforming both 1-layer and 3-layer variants. **Best 2** uses a similar 2-layer pipeline but swaps the sub-model composition to four temperature variants of LLaMA-3.3-70B-Instruct for Layer 1. Both surpass single LLaMA setups, underscoring MoA's ability to fuse multiple perspectives effectively.

Model	Setting	R-1	R-2	R-L	BLEU	MET	BS	AS	SC	Overall
LLaMA-3.3-70B-Instruct	Zero-shot	0.2476	0.0886	0.2156	0.0471	0.2777	0.8182	0.3096	0.2247	0.2786
	3-shot w/ H	0.2583	0.0968	0.2241	0.0487	0.2891	0.7612	0.2864	0.2345	0.2749
	3-shot w/ C	0.2733	0.0994	0.2398	0.0817	0.3055	0.8295	0.3151	0.2498	0.2993
	QLoRA SFT	0.2165	0.0778	0.1947	0.0315	0.2460	0.7960	0.2486	0.2033	0.2518
GPT-4o	Zero-shot	**0.4704**	<u>0.2340</u>	<u>0.4038</u>	**0.1307**	**0.4289**	**0.9116**	**0.4615**	**0.3031**	**0.4180**
	3-shot w/ H	<u>0.4519</u>	0.2291	0.3825	0.1193	0.3701	0.8821	0.4212	0.2543	0.3888
	3-shot w/ C	0.4515	**0.2524**	**0.4057**	<u>0.1212</u>	<u>0.3987</u>	<u>0.8901</u>	<u>0.4552</u>	0.2812	<u>0.4070</u>
MoA	Best 1	0.4372	0.2103	0.3611	0.1025	0.3305	0.8558	0.3913	0.2614	0.3688
	Best 2	0.4192	0.2055	0.3502	0.1096	0.3206	0.8512	0.3608	<u>0.2853</u>	0.3628

Table 2: **Task B**(perspective-based summarization) results. "3-shot w/ H(uman)" vs. "3-shot w/ C(lustering)" follows the same few-shot definitions as Table 1. Metrics include ROUGE (R-1, R-2, R-L), BLEU, METEOR (MET), BERTScore (BS), AlignScore (AS), SummaC (SC), and an Overall average.
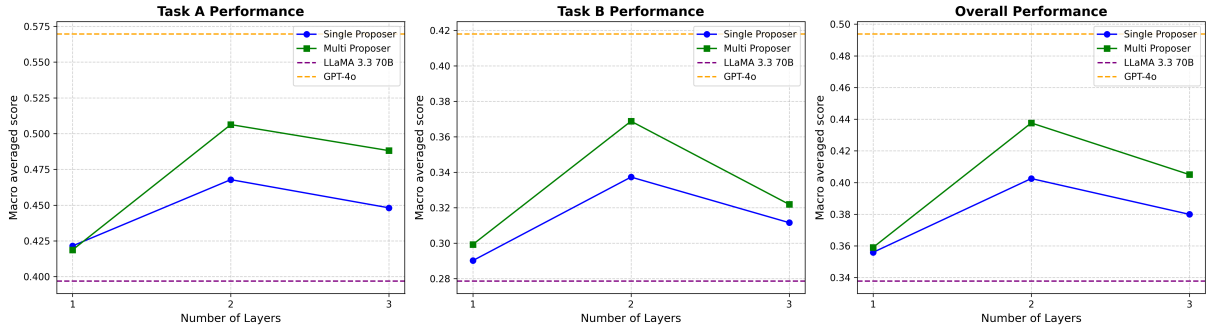

Figure 2: Performance comparison across different MoA layer counts.

MoA Best 1 at 0.5063, Best 2 at 0.4753 show strong improvements over single LLaMA-3.3-70B-Instruct baselines, though they still trail GPT-4o zero-shot. Nonetheless, MoA outperforms any single open-source LLM setting by a noticeable margin, 8% over LLaMA's best.

6.2 Task B: Perspective-Based Summaries

Table 2 shows the summarization performance, which is derived based on the best result spans from Task A, obtained using the optimal prompt detailed in Table 5 of Appendix C. Once again, the zero-shot GPT-4o approach leads with a general average of 0.4180, exceeding its 3-shot variants and aligning with the trends observed in Task A.

Few-shot Prompting For LLaMA-3.3-70B-Instruct, "3-shot w/ Clustering" yields 0.2993 overall vs 0.2749 with human-chosen examples and 0.2786 in zero-shot. Similarly for GPT-4o, sentence-transformer embedding based selection attains 0.4070, surpassing the human-chosen 3-shot (0.3888) while still slightly lower than the zero-shot GPT-4o (0.4180). Hence, while GPT-4o with zero-shot remains the single best, sentence-transformer embedding based few-shot tends to outperform manually curated exemplars.

MoA *Best 1* achieves 0.3688, while *Best 2* gets 0.3628, each notably exceeding LLaMA-3.3-70B-Instruct's best (0.2993). Although not rivaling GPT-4o, they confirm MoA's capacity to reduce hallucinations and unify multiple sub-model outputs.

QLoRA Supervised Fine-tuning For Task B, QLoRA fine-tuning yields an overall score of 0.2518 (Table 2), again lower than the corresponding zero- and few-shot results.

6.3 Ablation on Aggregators and Layering

Aggregator Comparison. Table 3 compares four aggregator models—LLaMA-3.3-70B-Instruct, Qwen-2.5-72B-Instruct, DeepSeek-R1-LLaMA-70B, GPT-4o-mini for the same MoA sub-model outputs (Best 1). LLaMA-3.3-70B-Instruct yields the highest Task A/B scores (0.5063 / 0.3688), while the GPT-4o mini aggregator drops to (0.4027 / 0.2981), showing that the aggregator choice is crucial.

Layering Comparison. Figure 2 illustrates how adding layers impacts MoA performance under two configurations:

- **Single Proposer:** Only LLaMA-3.3-70B-Instruct models are used to produce output

Aggregator	Task A	Task B
LLaMA-3.3-70B-Instruct	**0.5063**	**0.3688**
Qwen2.5-72B-Instruct	<u>0.4719</u>	<u>0.3456</u>
DeepSeek-R1-LLaMA-70B	0.4671	0.3411
GPT-4o-mini	0.4027	0.2981

Table 3: Performance comparison of different aggregators on Task A and Task B, holding the same MoA sub-model outputs as in "Best 1".

in each layer.

- **Multi Proposer:** LLaMA-3.3-70B-Instruct and Qwen-2.5-72B-Instrcut are combined to generate more diverse proposals.

In both cases, LLaMA-3.3-70B-Instruct is used as the aggregator, and the dashed lines indicate the zero-shot baselines (LLaMA: 0.3377 overall; GPT-4o: 0.4938 overall).

In the **Single Proposer** setting, the 1-layer model obtains an overall score of 0.3559, which increases to 0.4025 with 2 layers (a gain of 0.0466 points) but then drops to 0.3799 when using 3 layers. Similarly, in the **Multi Proposer** setting, the overall score rises from 0.3590 for 1 layer to 0.4376 for 2 layers (an improvement of 0.0786 points), before falling to 0.4050 with 3 layers.

These results indicate that adding a second layer consistently improves performance—yielding an improvement of roughly 14% over the LLaMA zero-shot baseline—while the third layer tends to over-correction, resulting in a performance drop. Thus, the 2-layer multi proposer configuration offers the best trade-off between enhancing overall accuracy and retaining valid outputs.

7 Conclusion

In this work, we addressed the challenge of perspective-aware summarization for healthcare Q&A. Our experiments show the recipe we tried and the final solution submitted for the challenge. With a bit disappointment, although MoA and embedding-based few-shot example selection improves the performance of open-source solution, the closed model, specifically GPT-4o in our case, still outperforms our best open-source solution by a large margin. Overall, our results highlight promising directions in leveraging large language models for multi-perspective healthcare Q&A, particularly when curated resources are scarce.

8 Limitations

Data size and quality could be one of major constraints. The generic training set might be too small to conduct effective finetuning. In our observation, Text span identification/classification annotations contain overlaps and ambiguities (e.g. extracted span starts with an incomplete word or punctuation), complicating the accuracy of perspective labels and gold summaries.

To apply an encoder-based model for span identification, we experimented with weighted NER finetuning (Appendix D). This approach assigns higher weights to underrepresented perspective categories to mitigate class imbalance. However, our results did not yield improvements, likely due to the inherent complexity and variability of user-generated content in the dataset. This suggests that alternative techniques, such as data augmentation or more robust fine-tuning strategies, may be necessary for handling imbalanced annotations effectively.

While MoA framework brings performance improvement, MoA configurations demand additional computational resources, especially in multi-layer or multi-agent setups.

Addressing these limitations, for example, through larger, more balanced datasets and more efficient aggregator layers, could further enhance perspective-aware summarization in real-world healthcare scenarios.

9 Future Work

To overcome current constraints, future endeavors could involve extracting more healthcare-related queries from broader corpora such as Natural Questions, followed by data augmentation via LLMs to create synthetic examples for underrepresented perspectives. A refined Mixture-of-Agents design could then integrate these enriched training sets for both classification and summarization tasks, thereby mitigating data scarcity, enhancing perspective coverage, and improving model generalizability across diverse healthcare topics.

Although our preliminary exploration shows that embedding-based selection boosts performance over manually curated exemplars, further studies on prompting construction techniques, like dynamic prompt construction (Gonen et al., 2022), retrieval-augmented prompting (Tang et al., 2025), or synthetic prompts (Kong et al., 2024), may lead to additional gains. We leave these investigations to future work, anticipating that such refinements will

further enhance the robustness and scalability of perspective-aware summarization in the healthcare domain.

Acknowledgments

References

Siddhant Agarwal, Md Shad Akhtar, and Shweta Yadav. 2025. Overview of the peranssumm 2025 shared task on perspective-aware healthcare answer summarization. In *Proceedings of the Second Workshop on Patient-Oriented Language Processing (CL4Health) @ NAACL 2025*, Albuquerque, USA. Association for Computational Linguistics.

Meta AI. 2024. Llama 3.3 model card and prompt formats. Accessed: 2024-12-06.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Ernie Chang, Xiaoyu Shen, Hui-Syuan Yeh, and Vera Demberg. 2021. On training instance selection for few-shot neural text generation. *ArXiv*, abs/2107.03176.

Rochana Chaturvedi, Abari Bhattacharya, and Shweta Yadav. 2024. Aspect-oriented consumer health answer summarization. *Preprint*, arXiv:2405.06295.

Tanya Chowdhury and Tanmoy Chakraborty. 2018. Cqasumm: Building references for community question answering summarization corpora. *Preprint*, arXiv:1811.04884.

Tanya Chowdhury, Sachin Kumar, and Tanmoy Chakraborty. 2020. Neural abstractive summarization with structural attention. *Preprint*, arXiv:2004.09739.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Preprint*, arXiv:2305.14314.

Alexander Fabbri, Xiaojian Wu, Srini Iyer, Haoran Li, and Mona Diab. 2022. AnswerSumm: A manually-curated dataset and pipeline for answer summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2508–2520, Seattle, United States. Association for Computational Linguistics.

Hila Gonen, Srini Iyer, Terra Blevins, Noah A. Smith, and Luke Zettlemoyer. 2022. Demystifying prompts in language models via perplexity estimation. In *Conference on Empirical Methods in Natural Language Processing*.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman,

and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Preprint*, arXiv:1506.03340.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Yilun Kong, Hangyu Mao, Qi Zhao, Bin Zhang, Jingqing Ruan, Li Shen, Yongzhe Chang, Xueqian Wang, Rui Zhao, and Dacheng Tao. 2024. Qpo: Query-dependent prompt optimization via multi-loop offline reinforcement learning. *ArXiv*, abs/2408.10504.

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. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Peter J. Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018. Generating wikipedia by summarizing long sequences. *Preprint*, arXiv:1801.10198.

Gauri Naik, Sharad Chandakacherla, Shweta Yadav, and Md Shad Akhtar. 2024. No perspective, no perception!! perspective-aware healthcare answer summarization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15919–15932, Bangkok, Thailand. Association for Computational Linguistics.

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.

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 Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, 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, Bogo Giertler, 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 Quinonero 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 Kon-

draciuk, 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 Aljubeh, 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. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li,

Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer. *Preprint*, arXiv:1910.10683.

Lance A. Ramshaw and Mitchell P. Marcus. 1995. Text chunking using transformation-based learning. *Preprint*, arXiv:cmp-lg/9505040.

Lei Tang, Jinghui Qin, Wenxuan Ye, Hao Tan, and Zhijing Yang. 2025. Adaptive few-shot prompting for machine translation with pre-trained language models. *Preprint*, arXiv:2501.01679.

Yongjian Tang, Rakebul Hasan, and Thomas Runkler. 2024. Fsponer: Few-shot prompt optimization for named entity recognition in domain-specific scenarios. *Preprint*, arXiv:2407.08035.

Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Zou. 2024. Mixture-of-agents enhances large language model capabilities. *Preprint*, arXiv:2406.04692.

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, Toronto, Canada. Association for Computational Linguistics.

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. *Preprint*, arXiv:1904.09675.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. 2024. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410, Bangkok, Thailand. Association for Computational Linguistics.

A Llama 3.3 70B QLoRA Supervised Fine-tuning Configs

As shown in Table 4, the following configuration was used for supervised fine-tuning using QLoRA for both Task A and Task B. The model was fine-tuned with 4 NVIDIA RTX A6000 GPUs. The only difference between the two tasks is the composition of the training dataset. This ensures that both tasks were fine-tuned under the same training environment, leveraging QLoRA to efficiently adapt the LLaMA-3.3-70B-Instruct model while maintaining computational efficiency.

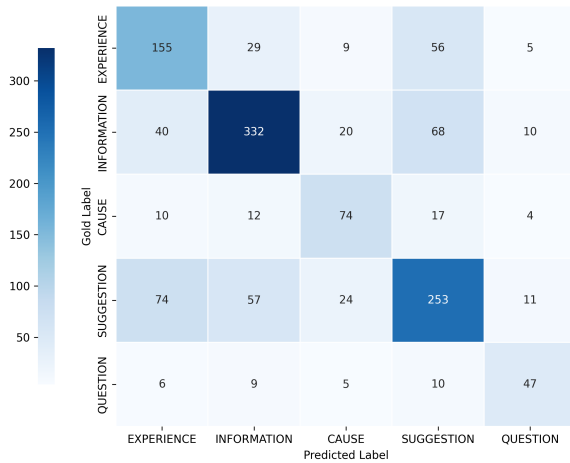## B Confusion Matrix for GPT-4o Zero-Shot

Figure 3: Confusion Matrix for GPT-4o Zero-Shot on Task A. Each cell indicates the number of samples in the corresponding gold-predicted label pair.

Figure 3 shows the confusion matrix (as a PNG image) for GPT-4o zero-shot on Task A (span classification). Rows correspond to the *gold* labels, and columns correspond to the *predicted* labels. Diagonal entries represent correctly classified samples for each perspective category, whereas off-diagonal entries indicate misclassifications (e.g., gold-labeled EXPERIENCE predicted as INFORMATION).

As illustrated in the confusion matrix, GPT-4o zero-shot achieves strong diagonal counts for each perspective label (EXPERIENCE, INFORMATION, CAUSE, SUGGESTION, QUESTION), indicating accurate predictions in most cases. The off-diagonal cells reflect scenarios where one perspective is mistaken for another, highlighting specific patterns of confusion (e.g., EXPERIENCE vs. INFORMATION). This strong performance aligns with our earlier

Parameter	Value
bf16	true
cutoff_len	3000
dataset	peranssumm_task
dataset_dir	data
ddp_timeout	180000000
do_train	true
double_quantization	true
eval_steps	5000
eval_strategy	steps
finetuning_type	lora
flash_attn	auto
gradient_accumulation_steps	2
learning_rate	5.0e-05
logging_steps	5
lora_alpha	16
lora_dropout	0.05
lora_rank	8
lora_target	all
lr_scheduler_type	cosine
max_grad_norm	1.0
max_samples	100000
model_name_or_path	{model_name}
num_train_epochs	3.0
optim	adamw_torch
output_dir	/path/to/output
packing	false
per_device_eval_batch_size	1
per_device_train_batch_size	1
plot_loss	true
preprocessing_num_workers	16
quantization_bit	4
quantization_method	bitsandbytes
report_to	none
save_steps	5000
stage	sft
template	llama3
train_on_prompt	true
trust_remote_code	true
val_size	0.3
warmup_steps	100

Table 4: QLoRA Supervised Fine-Tuning Configuration for LLaMA-3.3-70B-Instruct

quantitative results showing that GPT-4o zero-shot outperforms other baselines on span classification.

C Prompt Example

In Table 5, we present an example of the best prompt format for GPT-4o in zero-shot for both span identification/classification and perspective-

based summarization.

D NER Fine-tuning for Task A

In an exploratory experiment, we implemented a token-level BIO tagging(Ramshaw and Marcus, 1995) approach to perform span identification for Task A. In this method, each perspective is treated as a named entity with BIO labels (e.g., `B-INFORMATION`, `I-INFORMATION`, etc.), and the remaining tokens are tagged as `O`.

Data Preparation and Tagging. We first combined the question and answer texts and then tokenized the resulting sequence. Using the provided span annotations, we aligned token boundaries with the annotated spans to produce BIO tags. For instance, if an annotated span for the "CAUSE" perspective starts at character position s and ends at e, tokens falling entirely within this span are labeled as `B-CAUSE` for the first token and `I-CAUSE` for the subsequent tokens.

Class Weighting for Imbalance. To address class imbalance, we computed class weights as:

$$w_c = \frac{T}{n_c}, \quad \text{with } T = \sum_{c=1}^{C} n_c$$

where n_c denotes the total number of tokens belonging to class c, and T represents the total number of tokens across all classes. These weights were then incorporated into the cross-entropy loss function:

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^{N} w_{y_i} \log\left(\frac{\exp(z_{i,y_i})}{\sum_{c=1}^{C} \exp(z_{i,c})} \right)$$

where $z_{i,c}$ is the logit for token i and class c, and y_i is the ground-truth label.

Observations. Despite applying class weighting, our NER fine-tuning did not yield significant improvements. We attribute this to the small dataset size and the inherent challenge of labeling extended, overlapping spans—conditions that differ substantially from typical NER tasks involving shorter entity mentions. Consequently, while promising in principle, further investigation with larger or more targeted datasets is required.

Task	Span Identification and Classification	Perspective-Based Summarization
System Prompt	You are a helpful assistant.	
User Prompt	You are an expert annotator specialized in perspective-aware Healthcare Answer Summarization. First, validate that the document's content is aligned with the medical domain—ensure that it pertains to prevention, diagnosis, management, treatment of diseases, understanding of bodily functions, the effects of medications or medical interventions, or queries regarding wellness practices. Next, for each text span in the 'Answers' section, carefully assess and assign the most relevant perspective(s) from the following definitions: • **INFORMATION**: Knowledge about diseases, disorders, and health-related facts. • **CAUSE**: Reasons responsible for the occurrence of a medical condition. • **SUGGESTION**: Advice or recommendations to assist in making informed decisions. • **EXPERIENCE**: Individual experiences or anecdotes related to healthcare. • **QUESTION**: Inquiries for deeper understanding. Follow these instructions: • Only annotate spans from the 'Answers' section. • Ensure the document is medically relevant. • Multi-perspective labeling is allowed. • If a span explicitly mentions quantitative details, include that in your annotation. • Avoid personal bias and exclude links or personal identifiers. • Review your annotations to cover all underlying perspectives. Format your response as: `span: "<extracted text>", label: "<perspective>".`	While writing summaries, ensure that every essential idea and medical detail is captured from the extracted spans. Each summary should: • Be factually supported by the extracted spans. • Preserve all relevant insights and details. • Align clearly with the assigned perspective. • Avoid hallucinations, bias, or unverifiable content. Strictly adhere to the extracted spans to ensure factual consistency. Use the following structure for each perspective: • **INFORMATION**: "For information purposes, [summary]..." • **CAUSE**: "Some of the causes include [summary]..." • **SUGGESTION**: "It is suggested that [summary]..." • **EXPERIENCE**: "In user's experience, [summary]..." • **QUESTION**: "It is inquired whether [summary]..." Format your final summary as: `Summary: "<generated summary>"`.
Example Input	`{Question} + {Context} + {Answers} + {User Prompt}`	`{Question} + {Context} + {Spans} + {User Prompt}`

Table 5: Final prompt structure for Task A and Task B.