CIS 6200 Spring 2024, Project Report

Academic GPT: Leveraging Large Language Models for the Review of Machine Learning Research Papers

Team Members:

- Justin Gao; Email: gaoyukun@seas.upenn.edu
- Travis Xie; Email: yinuoxie@seas.upenn.edu
- Crescent Xiong; Email: zihanx3@seas.upenn.edu
- Chenghao Zhang; Email: zch719@seas.upenn.edu

Abstract

In the field of machine learning, the literature review process is crucial for maintaining the quality of academic conferences. This process, however, is challenged by the overwhelming number of submissions and a shortage of qualified reviewers. The surge in scholarly output and the increasing complexity of specialized knowledge further strain traditional review mechanisms, making high-quality peer reviews increasingly rare. This scarcity disproportionately affects junior researchers and those from under-resourced backgrounds, who may struggle to obtain timely and constructive feedback. Recent advancements in Large Language Models (LLMs), such as GPT-4, have highlighted their potential to automate the provision of scientific feedback on research manuscripts. In response, this project developed an automated pipeline using LLMs to review complete PDFs of machine learning papers. We utilized Chat-GPT 3.5, GPT-4, and a fine-tuned Mistral 7B model, which was trained on a corpus of 15,000 machine learning papers and their peer reviews. Our evaluation, based on a hit-rate metric, revealed that GPT-3.5 achieved a hit rate of 32%, GPT-4 reached 43%, and Mistral 7B scored 55%. Notably, Mistral 7B exhibited higher hit rates than GPT-3.5 in 84.21% of cases, and GPT-4 in 63.16% of cases. These findings suggest that LLMs, particularly when fine-tuned, can significantly aid the research review process. While there are accuracy limitations, LLMs can serve as valuable tools for researchers to preliminarily review their papers before submission. This could potentially improve acceptance rates and alleviate the burden on human reviewers by providing preliminary feedback, especially during the early stages of manuscript preparation.

1 Motivation

The escalating production of scholarly work coupled with the complexities of specialized knowledge presents unprecedented challenges in the academic research review process. This year's NeurIPS 2024 conference, for instance, received a record-breaking 13,321 submissions with only 1,596 reviewers available, averaging 8.3 papers per reviewer. This disproportionate ratio not only places significant burdens on reviewers, who often juggle their own research and additional conference duties, but also risks compromising academic quality and the dissemination of new ideas.

Despite the demonstrated capabilities of Large Language Models (LLMs) such as GPT-4 in generating human-like text across diverse domains, their potential to automate aspects of the scientific feedback process remains underexploited. These models can efficiently process vast amounts of text, potentially alleviating the burden on human reviewers by providing preliminary feedback. This feedback can help researchers refine their submissions before they undergo the traditional peer review process.

Our project is driven by the urgent need to enhance the scalability and accessibility of the academic review process. By leveraging LLMs, we aim to democratize access to valuable feedback, exploring the effectiveness of LLM-generated feedback in maintaining or even enhancing the quality of academic critiques. This initiative offers a unique opportunity to bridge the gap between the rising demand for peer reviews and the limited availability of reviewer resources.

Aligned with the study by Liang et al.[1], which showed that LLM-generated feedback on scientific papers could significantly overlap with human reviewer feedback and was found helpful by a majority of surveyed researchers, we explore the possibilities of fine-tuning an open-source model. Specifically, we focus on the Mistral 7B model[2], given

its advanced capabilities such as sliding window attention, which allows for incorporating long enough contexts, and its high performance across diverse benchmarks. These features make it an excellent base model for our task.

By developing and evaluating an automated pipeline (Figure 1) that uses LLMs to review complete PDFs of machine learning papers, this project not only seeks to enhance the peer review process but also aims to contribute to the broader discussion on integrating AI tools into scholarly communication. Through this work, we hope to provide empirical insights into the practical utility of LLMs in academic settings and pave the way for more sophisticated AI applications in the scientific review process.

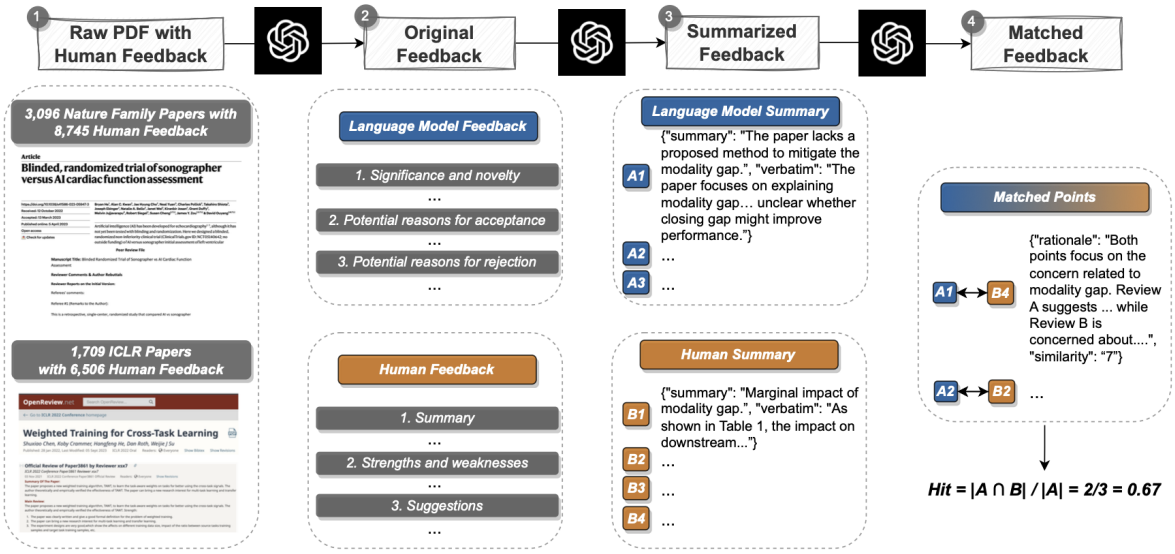

Figure 1: Workflow for retrospective analysis of LLM feedback. Inspired by Liang et al.[1], Our procedure systematically contrasts LLM generated feedback with human evaluations through a dual-stage pipeline. This involves an initial extractive summarization of comments from both LLM-generated and human feedback, followed by a semantic matching to identify overlapping commentary between LLM and human assessments.

2 Related Work

Large Language Models (LLMs) have demonstrated significant advancements across a variety of natural language processing tasks, including applications in domain-specific areas such as scientific writing and peer review. This section discusses prior research on the application of LLMs for scientific tasks, focusing on automating the peer review process and enhancing the capabilities of these models through efficient fine-tuning techniques.

Recent studies such as that by Liu and Shah[3] explore the utility of GPT-4 in performing specific tasks within the peer review process, including error detection and checklist verification. While their results affirm the model's aptitude for structured tasks, they also reveal its limitations in subjective evaluations like comparative assessments of paper quality. These insights underscore the necessity for advancements in model training to improve understanding and evaluative capabilities, which our project addresses through targeted fine-tuning techniques.

Further, Liang et al.[1] conducted a large-scale empirical analysis to assess the viability of GPT-4 in generating substantive scientific feedback. Their comparative analysis between GPT-4-generated feedback and human evaluations across a broad collection of scientific documents shows that while LLMs generally align with human input, there are notable disparities, especially in the depth and analytical rigor of the critiques. These findings are instrumental for our project as they guide our efforts to refine the depth and precision of LLM-generated feedback, aiming to elevate its utility to more comprehensive and contextually rich evaluations. Moreover, their development of an automated pipeline for generating and assessing paper reviews aligns with our goals of enhancing review efficiency and scalability.

The introduction of models like Galactica[4], which is trained on a vast corpus of scientific literature, and Academic GPT[5], which includes a component for paper review, showcases the potential of domain-specific LLMs.

However, these models do not focus extensively on peer review generation. In contrast, our work specifically fine-tunes an LLM to generate high-quality peer reviews, incorporating a comprehensive evaluation of the output.

Other research efforts have demonstrated the use of LLMs for tasks such as summarizing scientific papers[6] and extracting key information[7]. While these studies highlight the capabilities of LLMs in scientific domains, they do not directly address peer review generation.

Additionally, the development of Mistral 7B[2] introduces specialized capabilities crucial for handling extensive texts typical in research papers. Its superior performance, particularly in reasoning and code generation, makes it an ideal candidate for our project. By fine-tuning Mistral 7B on a dataset of machine learning papers, we aim to leverage its advanced features to generate more accurate and contextually relevant feedback.

In summary, our work builds upon the foundation of prior research on domain-specific LLMs and efficient fine-tuning techniques. We address the limitations of previous attempts to automate the peer review process by leveraging a carefully curated dataset, innovative preprocessing techniques, and a comprehensive evaluation framework. Our approach aims to provide a valuable tool for assisting researchers and reviewers in the peer review process while maintaining the quality and integrity of the scientific publication process.

3 Problem Formulation

Our study aims to harness the capabilities of Large Language Models (LLMs) for reviewing machine learning papers. To achieve this, we evaluate the efficacy of popular LLMs, opting to use GPT-3.5 and GPT-4 as baselines due to their widespread use and accessibility. These models were employed to generate reviews on both abstracts and complete texts of papers, with their performance compared to that of human reviews. Our methodology builds upon the pipeline utilized in the study by Liang et al. [1].

3.1 Baseline Configuration

3.1.1 Inputs

The inputs to our models consist of two distinct types of textual content from scientific papers:

- **Abstracts Only:** Only the abstract sections are provided to both GPT-3.5 and GPT-4 for a focused review.
- **Full Paper Texts:** The complete content of the papers, provided in PDF format.

The choice to use both abstracts and full paper texts as inputs for our models stems from a desire to understand the capabilities and limitations of LLMs in different review contexts. Abstracts provide a concise summary of the paper's aims, methods, and findings, allowing the models to generate focused reviews based on limited information. This can simulate scenarios where reviewers have restricted time or access to full texts, a common occurrence in real-world peer review processes. Conversely, full paper texts offer a comprehensive view of the research, enabling the models to generate more detailed and informed reviews. This comprehensive input allows us to assess the LLMs' ability to understand and critique the entirety of a scholarly work, including nuanced arguments and complex methodologies, providing a thorough comparison between abbreviated and full-text review capabilities.

3.1.2 Outputs

The models generate reviews of the following four criteria. We chose these criteria because they mirror the essential elements of traditional peer review, providing a comprehensive evaluation of a scientific paper's quality and relevance.

- **Significance and novelty:** This criterion assesses the importance and originality of the research within its field. It is crucial for determining whether the paper contributes new knowledge or solutions to existing problems, which is a primary factor in its scholarly impact and often dictates the interest and attention it will receive from the academic community.
- **Potential reasons for acceptance:** By summarizing reasons that may support the paper's acceptance, this output helps gauge the overall strength and quality of the research, including sound experimental design, robust data analysis, and clear alignment with the journal or conference themes. This aspect of the review helps highlight the paper's merits and readiness for publication.

- **Potential reasons for rejection:** Identifying and explaining possible flaws or shortcomings that could lead to a paper's rejection is essential for a balanced review. This criterion encourages a critical evaluation, focusing on weaknesses in methodology, gaps in the data, or logical inconsistencies that might undermine the research's validity or relevance.
- **Suggestions for improvement:** Providing actionable feedback to help authors enhance their work is a key function of peer review. This output not only aids in improving the current manuscript but also fosters the development of research skills and knowledge application for future projects. It ensures that the review process is constructive, offering practical advice that can lead to a higher chance of eventual acceptance.

Together, these criteria ensure that the model-generated reviews are multidimensional, addressing both the strengths and areas for development in a manner that is actionable and meaningful for authors and useful for editors and reviewers in making informed decisions.

The prompt used for generating LLM reviews is illustrated in Figure 2, and an example output is presented in the appendix (see Figure 8).

You are a professional machine learning conference reviewer who reviews a given paper and considers 4 criteria: [Significance and novelty], [Potential reasons for acceptance], [Potential reasons for rejection], and [Suggestions for improvement]. Please ensure that for each criterion, you summarize and provide random number of detailed supporting points from the content of the paper. And for each supporting point within each of criteria, use the format: '<title of supporting point>' followed by a detailed explanation.

The criteria you need to focus on are:

1. [Significance and novelty]: Assess the importance of the paper in its research field and the innovation of its methods or findings.

2. [Potential reasons for acceptance]: Summarize reasons that may support the acceptance of the paper, based on its quality, research results, experimental design, etc.

3. [Potential reasons for rejection]: Identify and explain flaws or shortcomings that could lead to the paper's rejection.

4. [Suggestions for improvement]: Provide specific suggestions to help the authors improve the paper and increase its chances of acceptance.

After reading the content of the paper provided below, your response should only include your reviews only, which means always start with [Significance and novelty], don't repeat the given paper and output things other than your reviews in required format, just extract and summarize information related to these criteria from the provided paper. The paper is given as follows:

Figure 2: Prompt template used to generate LLM reviews of the paper

3.2 Comparison Methods

To systematically compare LLM performance against human reviews, we organize the reviews into two distinct sets:

- **Set A (Reference Set):** These are the human-provided reviews, regarded as the gold standard.
- **Set B:** Reviews generated by LLMs from both full texts and abstracts.

Given the qualitative differences between LLM-generated reviews and human reviews, traditional metrics such as ROUGE are insufficient for our purposes. We therefore employ GPT-4 to facilitate a structured comparison through a two-stage methodology, each designed to mitigate specific challenges inherent in comparing human and machine-generated text.

3.2.1 Summarization Stage

The first stage involves using GPT-4 to condense human reviews into concise summaries that capture essential insights and critiques. This summarization is vital for establishing a baseline of comparison that matches the level of detail and abstraction typically provided by LLMs. This process helps to normalize the content across reviews, allowing for a more direct comparison in subsequent analysis stages. The prompt used for guiding this summarization process is illustrated in Figure 3.

Your goal is to identify the key concerns raised in the review, focusing only on potential reasons for rejection. Please provide your analysis in JSON format, including a concise summary, and the exact wording from the review.

Submission Title: {Title}

=====Review:

``` {Review Text} ```

=====

Example JSON format:

```
{{
    "1": {{"summary": "<your concise summary>", "verbatim": "<concise, copy the exact wording in the review>"}},
    "2": ...
}}
```

Analyze the review and provide the key concerns in the format specified above. Ignore minor issues like typos and clarifications. Output only JSON.

Figure 3: Prompt template used with GPT-4 for extractive text summarization of human and LLM feedback comments. This template is designed to ensure that GPT-4 extracts the most pertinent aspects of the reviews, focusing on synthesizing the core insights necessary for an equitable comparison in the following stage.

3.2.2 Cross-Comparison Stage

Following the summarization, the second stage involves a direct comparison between the summarized human reviews and the LLM-generated reviews. Here, GPT-4 is tasked with performing a semantic text matching to identify the degree of overlap and divergence between the two sets of reviews. This stage is crucial for evaluating the accuracy and completeness of LLM reviews in mirroring the substantive content of human reviews. The semantic matching process not only highlights the differences and similarities but also quantifies the effectiveness of LLMs in capturing nuanced human judgments. The prompt used for this comparison process is depicted in Figure 4.

3.3 Metrics

3.3.1 Similarity Score

We introduce a similarity scoring system in which each pair of reviews (from Set A and Set B) is evaluated for their concordance. This score quantifies the degree to which the LLM's interpretation of a paper aligns with the assessments of human reviewers. The objective is to measure the precision and agreement of the LLM's generated reviews with those considered as benchmarks in the field.

3.3.2 Threshold for Relevance

A predefined threshold of 7 out of 10 in the similarity scoring system is used to determine substantial agreement between human and LLM-generated reviews. Reviews achieving or surpassing this score are considered successful hits, indicating that the LLM has captured the essence of the human review to a significant extent. This threshold is established based on typical standards in qualitative content analysis, where a score of 70% or higher generally signifies strong agreement.

3.3.3 Hit Rate

The hit rate is calculated as the proportion of LLM-generated reviews that achieve the similarity threshold, providing a quantitative measure of the model's effectiveness in replicating human-like review quality. This metric is crucial for evaluating the practical utility of LLMs in academic peer review settings.

The metrics employed in our study were inspired by those proposed by Liang et al. [1]. In their research, the hit
```

```
Your task is to carefully analyze and accurately match the key concerns raised in two
    reviews, ensuring a strong correspondence between the matched points. Examine the
    verbatim closely.

=====Review A:
'''
<JSON extracted comments for Review A from previous step>
'''

=====Review B:
'''
<JSON extracted comments for Review B from previous step>
'''

Please follow the example JSON format below for matching points. For instance, if point 1
    from review A is nearly identical to point 2 from review B, it should look like this:

{{
"A1-B2": {{"rationale": "<explain why A1 and B2 are nearly identical>", "similarity":
    "<5-10, only an integer>"}},
...
}}


Note that you should only match points with a significant degree of similarity in their
    concerns. Refrain from matching points with only superficial similarities or weak
    connections. For each matched pair, rate the similarity on a scale of 5-10.

5. Somewhat Related: Points address similar themes but from different angles.
6. Moderately Related: Points share a common theme but with different perspectives or
    suggestions.
7. Strongly Related: Points are largely aligned but differ in some details or nuances.
8. Very Strongly Related: Points offer similar suggestions or concerns, with slight
    differences.
9. Almost Identical: Points are nearly the same, with minor differences in wording or
    presentation.
10. Identical: Points are exactly the same in terms of concerns, suggestions, or praises.


If no match is found, output an empty JSON object. Provide your output as JSON only.
```

Figure 4: Prompt template utilized with GPT-4 for performing semantic text matching to identify overlapping comments between two sets of feedback. The template structures inputs as two arrays of comments in JSON format, derived from the initial summarization stage. GPT-4 assesses and identifies commonalities, producing a JSON object where each key represents a pair of matched points, explaining the basis for their correlation.

rate, defined as the proportion of comments in Set A that correspond to those in Set B, is calculated as follows:

$$\text{Hit Rate} = \frac{|A \cap B|}{|A|}$$

3.3.4 Evaluation Pipeline

Figure 5 illustrates the overall workflow of the evaluation pipeline used to assess the similarity between comments from LLM-generated and human reviews. This workflow is divided into two stages:

1. **Extraction:** Utilizing the capabilities of LLMs for information extraction, key comments are systematically extracted from both LLM-generated and human-written reviews.

2. **Matching:** The LLM is employed for semantic similarity analysis, where comments from LLM and human feedback are compared. Each paired comment is assigned a similarity rating, with justifications provided for the

assigned scores. A similarity threshold of ≥ 7 is set to filter out weakly-matched comments, chosen based on empirical validations and the robustness of the matching stage demonstrated in preliminary tests.

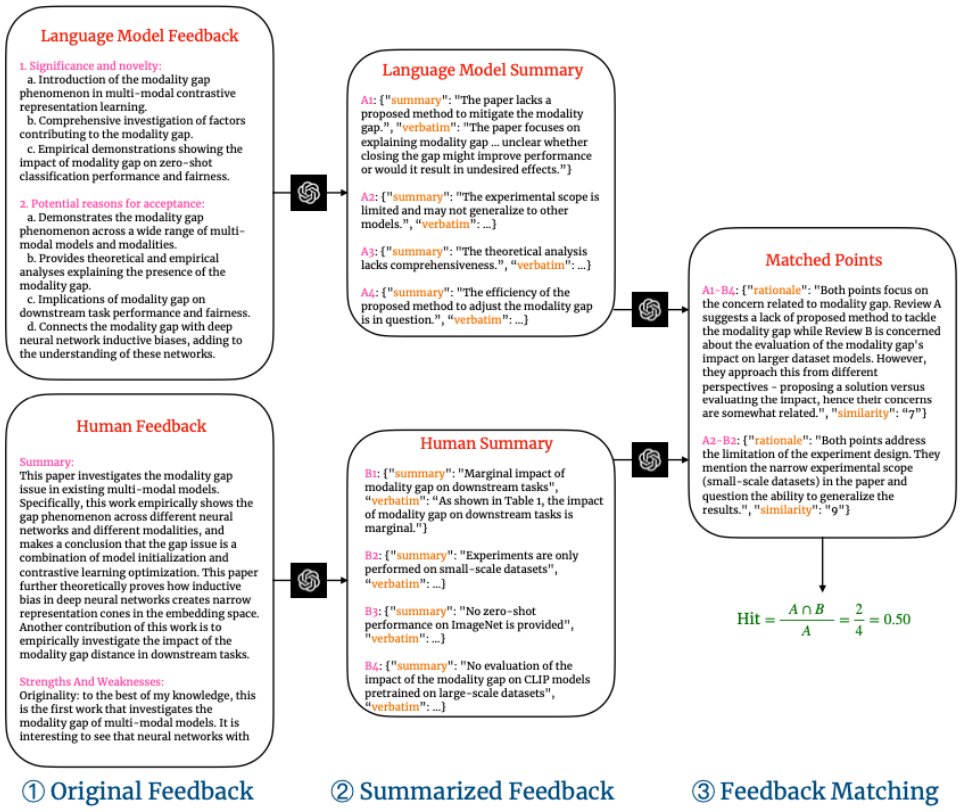

Figure 5: Workflow of the retrospective comment matching pipeline for scientific feedback texts. Panel 1 illustrates the overall two-stage comparison process. Panel 2 details the extraction of key comments using LLM capabilities, and panel 3 describes the semantic similarity analysis where matched comments are evaluated. A similarity threshold of ≥ 7 filters out comments with weak matches, ensuring only robust similarities contribute to the final analysis.

4 Methods

This section outlines our experimental design and methods employed to leverage the capabilities of LLM models for reviewing machine learning papers. In addition to applying GPT models directly, we explore the possibility of supervised fine-tuning an open-source LLM to better learn the task of reviewing machine learning papers.

4.1 Dataset Collection and Preprocessing

We collected a dataset comprising 15,566 paper-review pairs from the OpenReview platform. This dataset includes academic papers in PDF format alongside their corresponding human-written reviews. To prepare this data for training our model, we undertook several preprocessing steps.

4.1.1 PDF Parsing

Given that the academic papers were in PDF format, we needed a reliable PDF parser to extract the relevant textual content. We evaluated several open-source PDF parsers, including SciPDF Parser[8], ChatPaper Parser[9], Meta's "nougat" OCR tool[10], and ScienceBeam[8]. We selected SciPDF Parser as our primary tool based on its parsing accuracy, granularity, and computational efficiency.

SciPDF Parser, a variant of the classic GROBID parser, is particularly adept at extracting key components of academic papers such as titles, abstracts, figure and table captions, section titles, and main content. We enhanced the parsed output by inserting special tokens to demarcate different sections of the papers, like [TITLE], [ABSTRACT], [CAPTIONS], and [CONTENT]. This structured formatting aids the model's comprehension of the document layout.

4.1.2 Review Data Cleaning

The raw review data from OpenReview underwent extensive cleaning to enhance data quality and consistency. Our cleaning process included:

1. Removal of irrelevant content, such as author responses, acknowledgments, and non-peer-review comments.
2. Elimination of low-value information like literature references, author names, and publication dates, which do not contribute significantly to the review content.
3. Filtering out rare or inconsistent section headings to reduce noise and ensure uniformity across reviews.
4. Replacement of unsafe or ambiguous characters to avoid misinterpretation by the model.
5. Exclusion of excessively long or short reviews to maintain data consistency.
6. Rearrangement of review content to align with a Chain-of-Thought approach, improving the model's learning efficacy and user experience.

4.1.3 Review Aggregation and Summarization

Considering that each paper typically had multiple reviews, we used GPT-4 to aggregate and summarize these into a single coherent review. We crafted a prompt (Figure 6) that instructed GPT-4 to consolidate and organize key points from all reviews into the four main categories established in our baseline model. This structured approach facilitates a more unified and comprehensive analysis of feedback across different reviewers.

4.2 Model Selection and Training

4.2.1 Model Selection

For our experimental setup, we evaluated several candidate models specifically optimized for instruction-based tasks, including LLama2 7B Instruct[11], Gemma 7B Instruct[12], and Mistral 7B Instruct[2]. Our selection criteria were based on each model's demonstrated proficiency in handling complex NLP tasks and their ability to adhere to specific instruction sets, which are vital for generating coherent and contextually appropriate reviews.

Among the evaluated models, Mistral 7B-Instruct stood out due to its superior performance across various NLP benchmarks. Furthermore, the Gemma model, despite its potential, was not compatible with some of the advanced training methodologies we planned to implement. Consequently, we decided to proceed with the Mistral 7B-Instruct model, which had already been fine-tuned for similar tasks, thereby likely reducing the necessary training time and resource allocation.

4.2.2 Model Fine-tuning

Fine-tuning the selected Mistral 7B model proved to be a resource-intensive task. Initially, our aim was to fine-tune the model across the complete dataset of 15,566 paper-review pairs with a sequence length of 12,800 tokens—this length was chosen to cover the typical length of our dataset's contents, which ranged from 6,000 to 12,000 tokens. However, due to significant memory constraints encountered during the training process, we were compelled to modify our approach.

```
# input_prompt

Your task:
Compose a summary of some given reviews from a Machine Learning Conference written by reviewers. The given reviews is between the "######".

######
{Reviews}
######

You just need to use the following JSON format for output, **but don't output opinions that don't exist in the original reviews. As always, never guess why, if you're not sure, return an empty dict**:
{{
 'Significance and novelty': List multiple items by using Dict, The key is a brief description of the item, and the value is a detailed description of the item.
 'Potential reasons for acceptance': List multiple items by using Dict, The key is a brief description of the item, and the value is a detailed description of the item.
 "Potential reasons for rejection": List multiple items by using Dict, The key is a brief description of the item, and the value is a detailed description of the item.
 'Suggestions for improvement':  List multiple items by using Dict, The key is a brief description of the item, and the value is a detailed description of the item.
 }}
```

Figure 6: Example input prompt used with GPT-4 to aggregate and summarize multiple reviews into a single, coherent review, categorizing them into significance and novelty, potential reasons for acceptance, potential reasons for rejection, and suggestions for improvement.

To address the memory issues encountered during the fine-tuning of our large model, we implemented several advanced techniques known for their efficacy in optimizing training processes for deep neural networks.

QLora Technique The QLora method[13] is specifically designed to mitigate the high memory demands associated with training large-scale models. It achieves this by introducing quantization layers within the model architecture, which effectively reduces the precision of the numerical data used during computations. This reduction in data precision allows for a significant decrease in memory usage without substantially impacting the model's performance. By integrating QLora, we were able to maintain a balance between computational resource usage and model output quality, making it feasible to train on our extensive dataset under restricted memory conditions.

Enhancing Training Efficiency In addition to QLora, we incorporated several other methods to enhance the efficiency of the training process:

- **Accelerate:** As a high-performance library, Accelerate optimizes the execution of training tasks across multiple GPUs. It manages the distribution of computation tasks and data across the available hardware resources efficiently, maximizing throughput and minimizing idle times. This optimization is crucial for handling the high volume of data and complex computations involved in our project.

- **Flash Attention v2:** Introduced in the recent publication by Dao et al.[14], Flash Attention v2 is an advanced attention mechanism that significantly speeds up the computations involved in the attention layers of transformer models. By optimizing the way attention is calculated, Flash Attention v2 reduces the overall training time, enabling faster convergence of the model to high-quality solutions.

By integrating these methods into our training regimen, we were able to significantly reduce the training time while managing the memory limitations effectively. These optimizations ensured that we could leverage the full potential of the Mistral 7B-Instruct model under the constraints of our computational resources.

Despite these enhancements, the persistent memory limitations forced us to restrict our fine-tuning efforts to only the abstracts of the papers, reducing the sequence length to 1,024 tokens. This adjustment allowed us to manage the computational resources more effectively while still aiming to capture the critical elements of the reviews.

The training infrastructure was supported generously by Professor Ungar and CIS6200 excellent TAs, enabling us to utilize an AWS EC2 Instance equipped with 4 $\times$ A10G GPUs. This setup, running on the Deep Learning OSS Nvidia Driver AMI GPU PyTorch 2.2.0 (Amazon Linux 2), provided the necessary computational power to proceed with our training experiments. For those interested in replicating or extending our work, the hyperparameters and training code have been made available on our GitHub repository, accessible via this link.

5 Evaluation

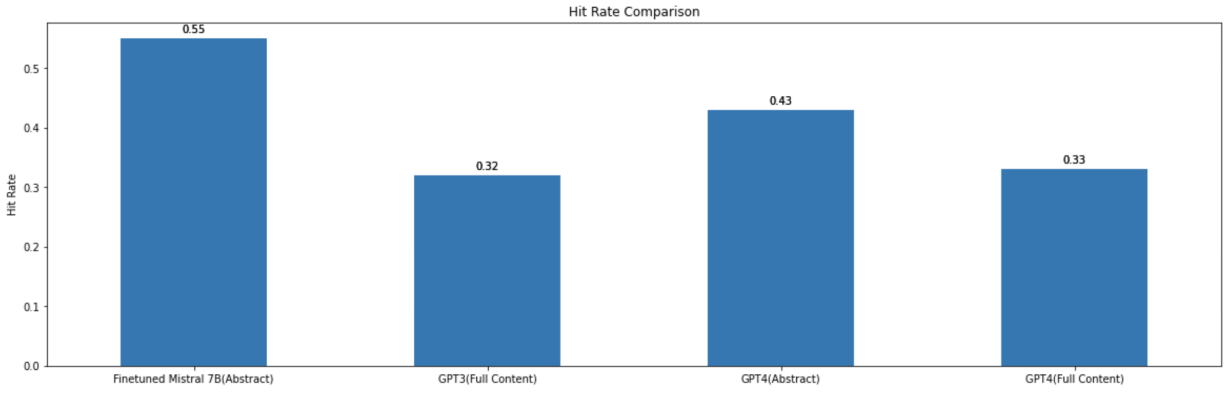

Figure 7: This bar chart compares the hit rates of different language models on two distinct tasks, abstract and full content processing. The chart displays that 'Finetuned Mistral 7B' has the highest hit rate for abstract tasks at 0.55. 'GPT3' shows a significantly lower performance with a hit rate of 0.32 for full content tasks. Meanwhile, 'GPT4' displays an improved hit rate of 0.43 for abstract tasks and a close 0.33 for full content tasks.

We evaluated various models on their analysis of selected computer science papers. Figure 7 presents the findings, showing that regardless of the scenario, the models' hit rate for analyzing abstracts was higher than for full content. This outcome suggests that abstract-based reviews may effectively retain the most salient points, aligning with human reviewers' assessments. On the other hand, the reduced hit rate for full-content reviews may imply a dilution effect, where the proliferation of details might lead to the omission of key aspects or the inclusion of less relevant information.

Additionally, our finetuned Mistral 7B performed significantly better than the others. Specifically, the finetuned Mistral 7B (Abstract) achieved a hit rate of 0.55 compared with GPT-3.5 Turbo's hit rate of 0.32. We have not yet conducted testing for finetuned Mistral 7B with full content due to memory space limitation issues.

References

[1] Weixin Liang, Yuhui Zhang, Hancheng Cao, Binglu Wang, Daisy Ding, Xinyu Yang, Kailas Vodrahalli, Siyu He, Daniel Smith, Yian Yin, Daniel McFarland, and James Zou. Can large language models provide useful feedback on research papers? a large-scale empirical analysis, 2023.

[2] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

[3] Ryan Liu and Nihar B. Shah. Reviewergpt? an exploratory study on using large language models for paper reviewing, 2023.

[4] Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*, 2022.

[5] Shufa Wei, Xiaolong Xu, Xianbiao Qi, Xi Yin, Jun Xia, Jingyi Ren, Peijun Tang, Yuxiang Zhong, Yihao Chen, Xiaoqin Ren, et al. Academicgpt: Empowering academic research. *arXiv preprint arXiv:2311.12315*, 2023.

[6] Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel S Weld. Tldr: Extreme summarization of scientific documents. *arXiv preprint arXiv:2004.15011*, 2020.

[7] Sarthak Jain, Madeleine Van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. Scirex: A challenge dataset for document-level information extraction. *arXiv preprint arXiv:2005.00512*, 2020.

[8] Scipdf parser: A python parser for scientific pdf based on grobid. `https://github.com/titipata/scipdf_parser`, 2018.

[9] Peter Gam Jiaxi Cui circlestarzero Shiwen Ni Jaseon Quanta Qingxu Fu Siyuan Hou Yongle Luo, Rongsheng Wang. Chatpaper: Use llm to summarize papers. `https://github.com/kaixindelele/ChatPaper`, 2023.

[10] Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. Nougat: Neural optical understanding for academic documents, 2023.

[11] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

[12] Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine

Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. Gemma: Open models based on gemini research and technology, 2024.

[13] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms, 2023.

[14] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning, 2023.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

6 Appendix

```
[Significance and novelty]
The "Transformer" paper introduces a novel architecture that challenges the dominant sequence transduction models
which rely on recurrent neural networks (RNNs) and convolutional neural networks (CNNs). The main contributions an
d innovations of this work include the introduction of an architecture based entirely on attention mechanisms, eli
minating the need for recurrence or convolution within the model, which represents a significant shift in the appr
oach to machine translation and potentially other sequence modeling tasks.

- 'Elimination of recurrence and convolution': The proposed Transformer model is novel in its exclusive reliance o
n attention mechanisms, avoiding traditional RNNs and CNNs which can be computationally expensive and slow due to
their inherent sequential nature.
- 'Optimization for Parallelization': The Transformer is designed to maximize parallel computational capabilities,
reducing training time significantly compared to existing models.
- 'Introduction of Multi-Head Attention': This paper is the first to implement Multi-Head Attention, allowing the
model to jointly process information from different representation subspaces at different positions, which enhance
s the model's ability to capture complex dependencies in the data.

[Potential reasons for acceptance]
The Transformer model demonstrates state-of-the-art performance on standard machine translation tasks, significant
improvements in training efficiency, and detailed experimental analysis, all of which strongly support the accepta
nce of this paper.

- 'Superior Performance': The model outperforms all previous models, including ensembles, on the WMT 2014 English-
to-German and English-to-French translation tasks, with substantial improvements in BLEU scores.
- 'Efficiency in Training': It is noted that the Transformer requires substantially less time to train—up to 3.5 d
ays on eight GPUs, making it more practical and accessible for use in different settings.
- 'Extensive Experimental Validation': The authors provide a comprehensive set of experiments and ablations that d
emonstrate the effectiveness of different components of the Transformer, such as variations in the number of atten
tion heads and the use of sinusoidal vs. learned positional encodings.

[Potential reasons for rejection]
Despite its strengths, there are a few aspects that might concern reviewers, mainly related to the generalizabilit
y and the computational demand of the model.

- 'Heavy Computational Resources': While more efficient than some prior models, the Transformer still requires sig
nificant computational power (e.g., multiple GPUs), which might not be feasible for all research or practical appl
ications.
- 'Limited Evaluation Scope': The application of the Transformer is demonstrated primarily on machine translation
tasks. Its performance and utility on other NLP tasks or in non-NLP domains remain unexplored in this paper.
- 'Potential Overfitting in Smaller Datasets': The model's complexity and capacity might lead to overfitting when
applied to smaller datasets or less resource-rich languages, a point that isn't addressed thoroughly.

[Suggestions for improvement]
To strengthen the paper further, the authors might consider addressing the generalizability of the Transformer mod
el and experiment with computational optimizations.

- 'Broader Evaluation on Various Tasks': Expanding the evaluation to include a variety of other tasks such as summ
arization, text generation, and beyond NLP tasks can demonstrate the adaptability and broad utility of the Transfo
rmer model.
- 'Investigation into Model Compression': Research into model compression techniques or more efficient attention m
echanisms could help in reducing the computational load, making the Transformer feasible on lower-resource setting
s.
- 'Experiments on Smaller and Diverse Datasets': Additional experiments on smaller or low-resource datasets could
help understand the model's performance limitations and improvements in handling overfitting.
```

Figure 8: GPT-4 reviews on Attention is all you need[15] paper