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

The rapid increase in scholarly output presents significant challenges in the peer review process due to the limited availability of qualified reviewers and the growing complexity of machine learning research. This project introduces an automated review pipeline using Large Language Models (LLMs), specifically leveraging ChatGPT 3.5, GPT-4, and a fine-tuned Mistral 7B model, to assess the feasibility of automating the provision of feedback on machine learning research papers. We developed and evaluated a dual-pipeline system: one for generating reviews from complete PDFs of research papers and another for evaluating these reviews against human-generated feedback. Our approach leverages the strengths of LLMs to handle extensive texts and generate comprehensive, insightful reviews, aiming to alleviate the strain on human reviewers by supplementing the conventional review process. The system's effectiveness was validated through comparative analysis using overlap metrics like the hit-rate, Szymkiewicz–Simpson Overlap Coefficient, the Jaccard Index, and the Sørensen–Dice Coefficient, demonstrating that LLM-generated feedback could closely approximate human-like critique, thus supporting the potential integration of LLMs into the peer review workflow. These results underscore the potential for LLMs to significantly enhance the scalability and accessibility of the peer review process, potentially revolutionizing academic publishing by providing timely, reliable, and scalable reviews. Our project is avaiable at our Github repository.

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.

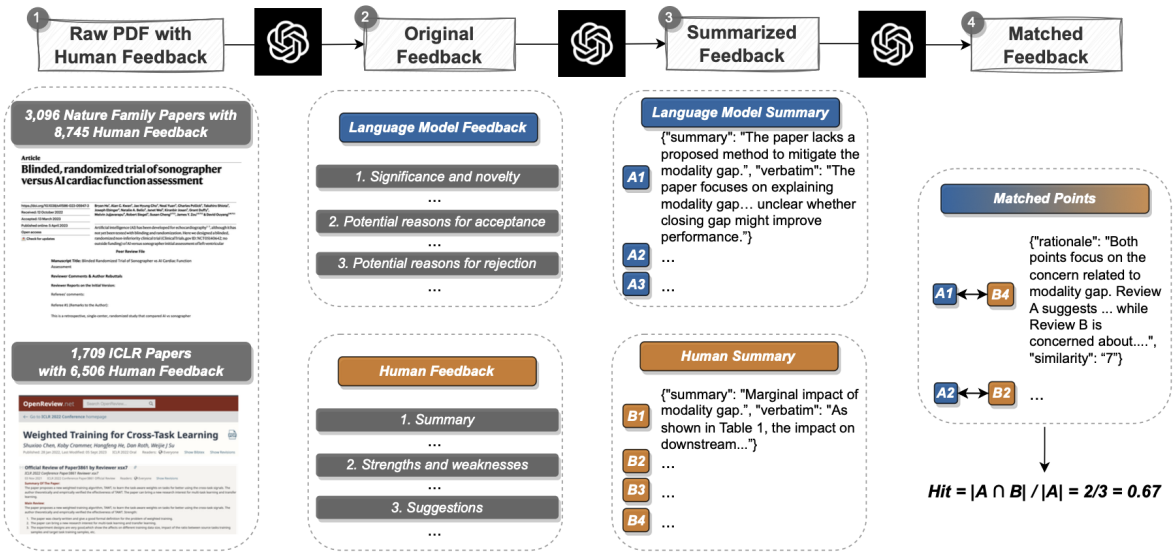

Figure 1: Workflow for retrospective analysis of LLM feedback. Inspired by Liang et al.[1], Our procedure systematically contrasts LLM generated feedback with human evaluations through a dual-stage pipeline. This involves an initial extractive summarization of comments from both LLM-generated and human 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 is consisted of two pipelines, the first one is the generation pipepline, which intends to generate reviews applying the baseline model, and the second one is the evaluation piepline, which eveluates the LLM generated review by comparing it with the human reviews.

3.1 Review Generation Pipeline

The generation pipeline is described in Figure 2. This pipeline utilizes a parser to extract structured data from raw PDF files and subsequently feeds both the prompt and the extracted data into a model to generate reviews.

3.1.1 PDF Parser

Given that academic papers are typically in PDF format, a reliable PDF parser was essential for extracting 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]. SciPDF Parser was selected as our primary tool based on its accuracy, granularity, and computational efficiency.

SciPDF Parser, an enhanced variant of the classic GROBID parser, excels at extracting key components such as titles, abstracts, figure and table captions, section titles, and main text. We improved the parser's output by inserting special tokens to demarcate different sections of the documents, like [TITLE], [ABSTRACT], [CAPTIONS], and [CONTENT], enhancing the model's ability to understand the document layout.

3.1.2 Inputs

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

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

The use of both abstracts and full paper texts as inputs allows us to evaluate the capabilities and limitations of large language models (LLMs) in different review contexts. Abstracts, which summarize the aims, methods, and findings of a paper, allow for focused reviews under constraints similar to those found in real-world peer review processes. Full texts, providing a comprehensive view of the research, enable the models to generate detailed and informed reviews.

The structured inputs extracted from the raw PDF files are later included in our prompts to generate reviews. The prompt used for generating LLM reviews is illustrated in Figure 3.

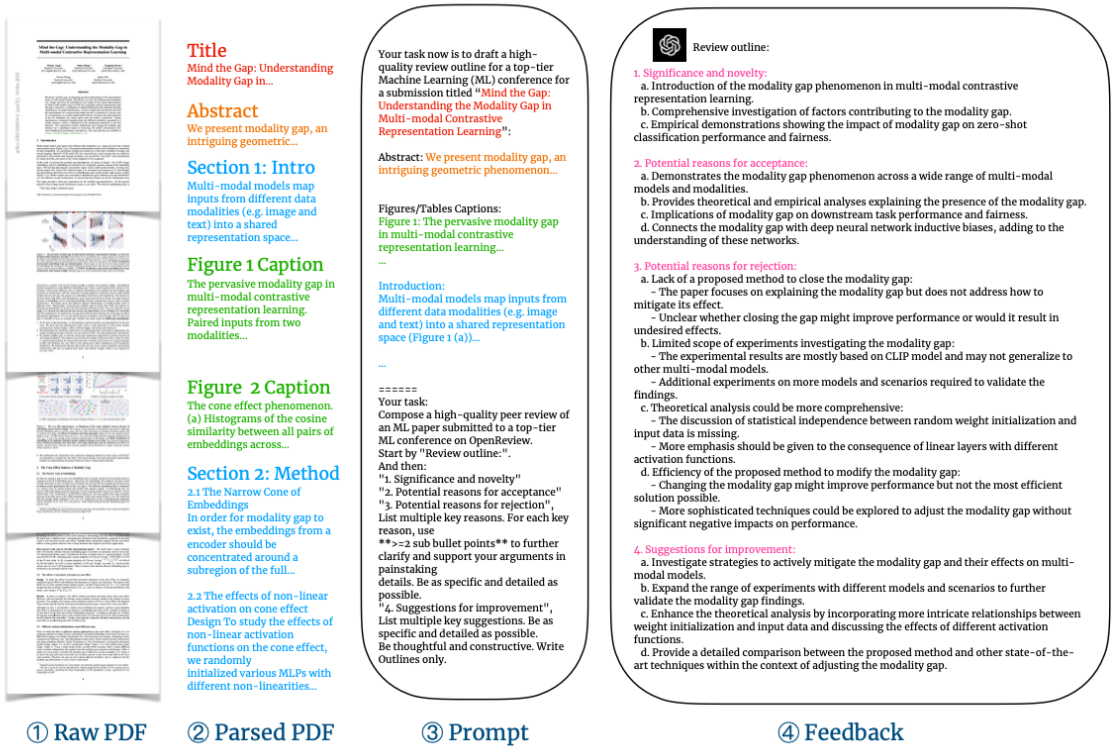

Figure 2: Workflow of the review generation pipeline. This pipeline utilizes a parser to extract structured data from raw PDF files and subsequently feeds both the prompt and the extracted data into a model to generate reviews.

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 3: Prompt template used to generate LLM reviews

3.1.3 Outputs

The models generate reviews based on the following four criteria, which mirror the essential elements of traditional peer review:

- **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.

These criteria ensure that the model-generated reviews are multidimensional and actionable, meaningful for authors and useful for editors and reviewers in making informed decisions. An example of the output is presented in the appendix Figure 10.

3.2 Evaluation Pipeline

To systematically assess LLM performance against human reviews, we categorized the reviews into two distinct sets:

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

Traditional metrics like ROUGE are inadequate for comparing qualitative aspects of LLM-generated reviews with human reviews. Thus, we employ a two-stage methodology using GPT-4 to facilitate a structured comparison, addressing the inherent challenges in comparing human and machine-generated texts.

Figure 4 illustrates the evaluation pipeline, divided into two stages: summarization and matching.

3.2.1 Summarization Stage

In the first stage, GPT-4 condenses the key points from both human and LLM reviews into concise summaries, capturing essential insights and critiques. This process standardizes the content across different reviews, facilitating a direct comparison in the matching stage. The guiding prompt for this summarization is shown in Figure 5.

3.2.2 Matching Stage

The second stage involves semantic text matching between the summarized reviews from Set A and Set B, using GPT-4. Each pair of reviews is assessed for semantic similarity, quantified by a scoring system. A threshold of 7 out of 10 is used to filter out weak matches, ensuring only significant similarities contribute to the analysis. The comparison prompt is depicted in Figure 6.

A similarity score of 7 or higher is deemed substantial, indicating that the LLM's review captures the essence of the human review effectively. This threshold is established based on standards typical in qualitative content analysis, where a 70% agreement generally signifies strong concordance.

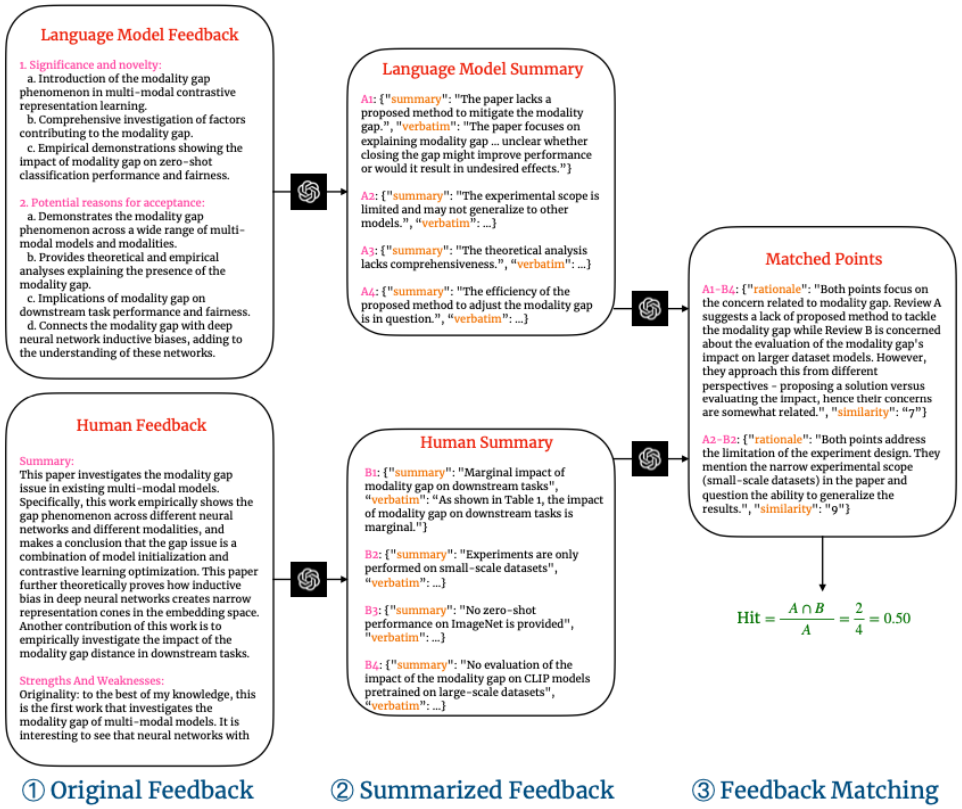

Figure 4: Workflow of the evaluation pipeline. Panel 1 illustrates the two-stage comparison process; Panel 2 details the extraction of key comments; Panel 3 describes the semantic similarity analysis, including a similarity threshold of ≥ 7 to ensure robust matches.

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 5: Prompt template for GPT-4 extractive text summarization, designed to synthesize core insights for equitable comparison in the subsequent matching stage.
```

```
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 6: Prompt template for semantic text matching, structuring inputs as JSON arrays of comments. This process assesses overlaps and commonalities, producing a JSON object explaining the correlation basis for each matched pair.

3.2.3 Metrics

To assess the performance of our mode, we employed four evaluation metrics - hit rate, Jaccard index, Sorensen-Dice coefficient, and Szymkiewicz-Simpson coefficient - to comprehensively evaluate the similarity between the model-generated reviews and the benchmark human reviews.

The Hit rate measures the proportion of model-generated reviews that match the human reviews, providing a high-level indication of overall performance. It is defined as:

$$\text{Hit Rate} = \frac{|A \cap B|}{|A|} \tag{1}$$

The Jaccard index, also known as the intersection over union (IoU), calculates the size of the intersection divided by the size of the union of the review sets. It is defined as:

$$\text{Jaccard}(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{2}$$

where A and B are the sets of elements in the model-generated and human reviews, respectively.

The Sorensen-Dice coefficient is similar to the Jaccard index but gives more weight to the overlapping elements. It is computed as:

$$\text{Dice}(A, B) = \frac{2 * |A \cap B|}{|A| + |B|} \tag{3}$$

The Szymkiewicz-Simpson coefficient, also called the overlap coefficient, measures the overlap between two sets and is defined as:

$$\text{Overlap}(A, B) = \frac{|A \cap B|}{\min(|A|, |B|)} \tag{4}$$

4 Methods

We explored two two additional experimental methods in addition to the baseline model.

1. **One-shot Learning:** We utilized the one-shot learning capabilities of GPT-3.5 and GPT-4, applying them to "abstract only" inputs. This method tests how well these models can perform when provided with a detailed example within the constraints of limited context lengths.

2. **Fine-Tune Mistral 7B Instruct Model:** We explored supervised fine-tuning of an open-source LLM, specifically the Mistral 7B Instruct Model, to better adapt it for the task of reviewing machine learning papers.

Details on these methods will be discussed after outlining the dataset collection process.

4.1 Dataset Collection and Preprocessing

We compiled a dataset consisting of 15,566 paper-review pairs sourced from the OpenReview platform. This collection includes academic papers in PDF format and their corresponding human-written reviews, as demonstrated in Figure 7 below.

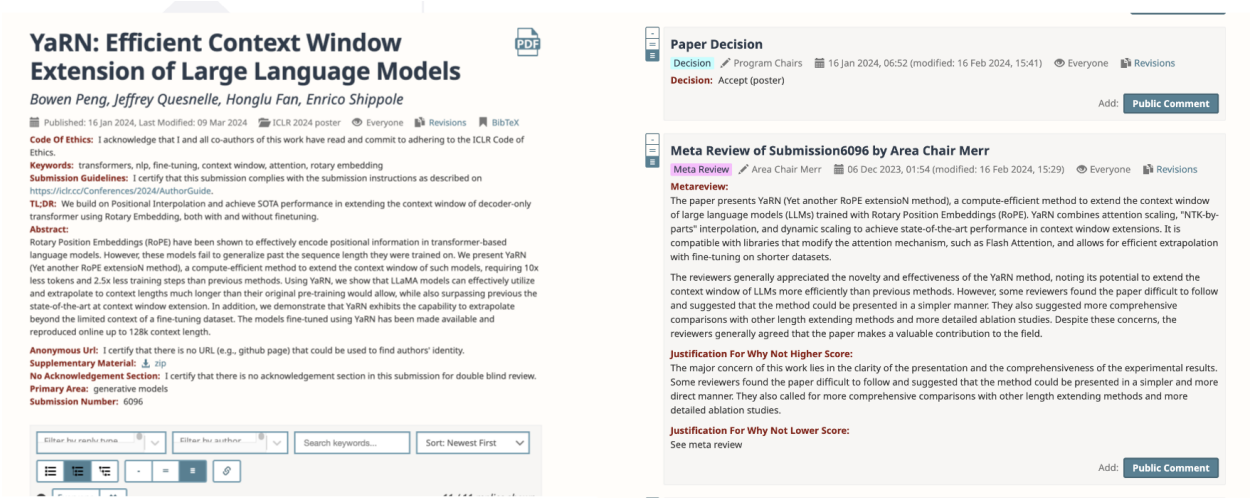

Figure 7: An example of a paper and its reviews from the OpenReview platform.

The dataset underwent extensive preprocessing to prepare it for training our models.

4.1.1 Review Data Cleaning

The raw review data required significant cleaning to improve quality and consistency:

1. Removal of irrelevant content such as author responses, acknowledgments, and non-peer-review comments.

2. Elimination of extraneous information like literature references, author names, and publication dates, which are not pertinent to the review's substance.

3. Standardization by filtering out rare or inconsistent section headings, reducing noise and ensuring uniformity.
4. Replacement of unsafe or ambiguous characters to prevent misinterpretation by the model.
5. Exclusion of excessively lengthy or brief reviews to maintain data consistency.
6. Structuring review content to align with a Chain-of-Thought approach, enhancing the model's learning efficacy and user experience.

4.1.2 Review Aggregation and Summarization

Each paper typically received multiple reviews. We utilized GPT-4 to aggregate and summarize these into a single coherent review. The prompt shown in Figure 8 instructed GPT-4 to consolidate key points from all reviews into four main categories defined in our baseline model, providing a unified and comprehensive analysis of feedback from different reviewers. This structured approach is crucial for training a consistent model.

```
# 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 8: 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.

4.2 One-Shot Learning

To evaluate the one-shot learning effectiveness of GPT-3.5 and GPT-4, we provided an example of a paper's abstract along with its aggregated reviews. This setup was designed to assess the models' capability to generate insightful reviews based on a single detailed example under constrained input conditions. The primary goal was to explore the limits of in-context learning, particularly how effectively the models could extrapolate from one example to generalize over unseen texts in a similar domain.

The process involved feeding the models an abstract as input, followed by a detailed, model-generated review of a similar paper as context. This was intended to simulate the potential real-world application of these models in academic settings where detailed peer reviews are crucial yet time-intensive. We hypothesized that by demonstrating a high-quality review example, the models would align their subsequent outputs more closely with the standards observed in human-written reviews.

4.3 Model Fine-Tuning

4.3.1 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.

4.3.2 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.

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 Experiments and Results

5.1 Experiments

To evaluate the performance of our model, we conducted a systematic analysis using a randomly selected subset of 100 papers from our database. We applied the evaluation pipeline previously described to generate hit-rate metrics for these papers. This approach allowed us to quantitatively assess how well our model's generated reviews aligned with the benchmark human reviews.

For a more granular inspection and to facilitate reproducibility, we also prepared a toy dataset comprising 20 papers and their corresponding review results. This dataset is intended to provide a snapshot of the model's performance on a smaller scale, offering insights into its operational dynamics under controlled conditions. The dataset, along with the review results, can be accessed at ourGithub.

5.2 Results

Figure 9 presents the results of our comparative analysis. The fine-tuned Mistral_reviews model achieves a hit rate of 0.324, outperforming both GPT-3.5_abstract_reviews_one_shot (0.302) and GPT-4_full_reviews (0.296). However, GPT-3.5_full_reviews demonstrates the highest hit rate at 0.549.

The Jaccard index and Sorensen-Dice coefficient follow a similar trend, with our model surpassing the GPT-4_full_reviews but falling slightly short of GPT-3.5_abstract_reviews. The Szymkiewicz-Simpson coefficient shows a closer performance gap between our model and 3.5_abstract_reviews, indicating a higher degree of overlap in the generated reviews.

These results suggest that our fine-tuned Mistral-7B-Instruct-v0.2 model is capable of generating reviews that closely resemble human-written ones, demonstrating competitive performance against the GPT-4 models. GPT-3.5_abstract_reviews maintains an edge in terms of overall similarity to the benchmark reviews. However, that doesn't mean GPT 3.5 actually perform better, as discussed in the Analysis Section.

Model	Hit Rate	Jaccard Index	Sorensen Dice Coefficient	Szymkiewicz-Simpson Coefficient
GPT 3.5:				
Full Reviews	**0.549**	0.327	0.451	0.544
Abstract Reviews	0.513	**0.361**	**0.470**	**0.592**
One-Shot	0.302	0.262	0.356	0.450
GPT 4:				
Full Reviews	0.296	0.216	0.275	0.317
Abstract Reviews	0.410	0.292	0.405	0.475
One-Shot	0.346	0.267	0.371	0.450
Mistral:				
Original	0.463	0.318	0.440	0.508
Fine-Tuned(ours)	0.324	0.239	0.333	0.467

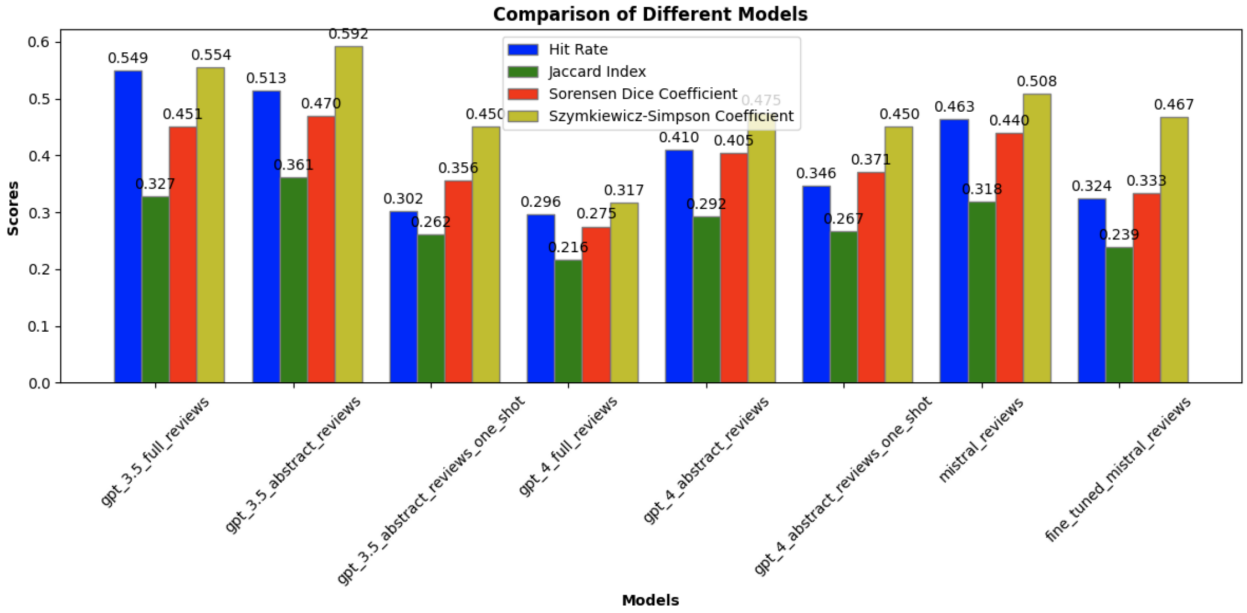

Figure 9: Performance comparison of different models for paper review generation. The evaluated models include our fine-tuned Mistral-7B-Instruct-v0.2 model, GPT-3.5 Turbo (using abstract and full paper), and GPT-4 (using full paper). Four metrics are used for evaluation: hit rate, Jaccard index, Sorensen-Dice coefficient, and Szymkiewicz-Simpson coefficient. Counterintuitively, GPT-3.5 with abstract reviews(zero shot) perform the best performace.

5.3 Analysis

The results of our comparative study reveal some interesting insights into the performance of different language models for generating paper reviews. Surprisingly, GPT-3.5 models, especially GPT-3.5 Abstract Reviews with Zero-shot, outperform our fine-tuned Mistral model and GPT-4 variants in terms of hit rate and Szymkiewicz-Simpson coefficient.

One possible explanation for GPT-3.5's superior performance is its tendency to generate more general reviews that apply to a broad range of papers. This characteristic may result in higher scores, as the generated reviews are more likely to align with the benchmark human reviews, but without more specific reviews. Interestingly, the addition of one-shot learning to GPT-3.5 leads to a significant performance drop. This may suggests that more critical and specific reviews generated by our fine-tuned Mistral model, GPT-4, or GPT 3.5 with One-Shot may not always achieve higher scores, as they might deviate from the general reviews.

Another factor contributing to the lower performance of our fine-tuned Mistral model is the occasional occurrence of hallucination problems. Due to limited training resources and dataset size, our model may generate reviews that include irrelevant or incorrect information, which negatively impacts its scores.

To illustrate the differences between the models, let's consider an example. For the paper "Evaluating Cognitive Maps and Planning in Large Language Models with CogEval"by Momennejad et al.[15], our fine-tuned Mistral model generates the following suggestions for improvement:

> The reviewers suggest improving the paper's in 4 out-of-the box planning abilities, utilizing more precise measurements and controls to assessing the cognitive maps, and providing more detailed explanations of the planning performance of language models. The reviewers also encourage the authors to include evaluation in areas such as text-to-speech generation, natural language inference, and other tasks beyond the 8 planning abilities.

In contrast, even GPT-4 will provide a more general set of revews:

> 'Enhance comparative analysis': The study could be strengthened by including comparisons with other evaluation methods or benchmarks in the field. This would help in positioning CogEval's effectiveness and novelty more explicitly.

'Broaden the range of LLMs and tasks': Expanding the scope to include a wider variety of LLMs and additional cognitive tasks could help in generalizing the findings. This would also address potential criticisms regarding the specific selection of LLMs and tasks.

'Simplify presentation of methods and results': Streamlining the presentation of the experimental methodologies and findings, possibly through clearer visualizations and less technical jargon, could make the paper more accessible to a broader audience.

'Address potential biases and limitations': The paper would benefit from a deeper discussion on the potential biases in task design and model selection. Adding more on how these factors might influence the generalizability of the results could enhance the paper's robustness.

While GPT-4's output seems more comprehensive and addresses a wider range of aspects, such as comparative analysis, generalizability, presentation clarity, and potential biases, our fine-tuned Mistral model generates more concise and specific suggestions.

6 Conclusion and Discussion

In this project, we aimed to leverage Large Language Models (LLMs) for reviewing machine learning research papers, with the goal of enhancing the efficiency and accessibility of the academic review process. By developing and evaluating an automated pipeline using GPT-3.5, GPT-4, and a fine-tuned Mistral 7B model, we explored the potential of LLMs in generating high-quality peer reviews.

Our comparative analysis revealed that while GPT-3.5 models achieved higher scores in terms of hit rate and overlap coefficients, their performance may not necessarily reflect the quality of the generated reviews. The superior scores of GPT-3.5 can be attributed to its tendency to generate more general reviews that apply to a broader range of papers. In contrast, our fine-tuned Mistral model and GPT-4 variants generated more specific and critical reviews, which may not always align perfectly with the benchmark human reviews but provide more valuable insights for authors.

The occasional occurrence of hallucination problems in our fine-tuned Mistral model, due to limited training resources and dataset size, highlights the need for further improvement in model training and optimization. Increasing the dataset size and diversity, as well as exploring more advanced fine-tuning techniques, could help mitigate these issues and enhance the model's ability to generate accurate and relevant reviews.

Moreover, our findings underscore the importance of developing more comprehensive and nuanced evaluation metrics for assessing the quality of LLM-generated reviews. While our current metrics provide a quantitative assessment of similarity between model-generated and human reviews, they may not fully capture the depth, specificity, and constructive nature of the feedback. Future research should focus on designing evaluation frameworks that better reflect the qualitative aspects of peer reviews, such as the actionability of suggestions and the identification of key strengths and weaknesses.

Despite these challenges, our project demonstrates the potential of LLMs in automating and augmenting the academic review process. By generating preliminary feedback and assisting human reviewers, LLMs can help alleviate the burden on reviewers and improve the accessibility of valuable feedback for researchers. As LLM capabilities continue to advance, we anticipate that their role in scholarly communication will expand, ultimately contributing to a more efficient and inclusive academic ecosystem.

Future steps in this line of research may include:

1. Developing more sophisticated evaluation metrics that better capture the qualitative aspects of peer reviews, such as the depth, specificity, and constructive nature of the feedback.

2. Expanding the training dataset to include a wider range of research domains and paper types, enabling the model to generate more diverse and domain-specific reviews.

3. Exploring more advanced fine-tuning techniques, such as multi-task learning and transfer learning, to further improve the model's performance and generalization capabilities.

4. Investigating the potential of LLMs in other aspects of the academic review process, such as assisting with editorial decisions, detecting plagiarism, and identifying potential reviewers based on expertise.

5. Conducting user studies to assess the perceived value and usability of LLM-generated reviews from the perspective of authors, reviewers, and editors.

In conclusion, our project represents an important step towards harnessing the power of LLMs for enhancing the academic review process. By addressing the limitations identified in our study and continuing to refine LLM capabilities, we believe that these models can play a transformative role in scholarly communication, ultimately fostering a more efficient, inclusive, and collaborative research ecosystem.

Acknowledgments

We extend our deepest gratitude to Professor Lyle Ungar for his invaluable guidance of the CIS6200 course and support throughout the process of this project. His expertise and insights have been fundamental to the success of our project, providing us with direction and encouragement at every stage.

We are also immensely grateful to our teaching assistants, Visweswaran Baskaran, Haotong (Victor) Tian, and Royina Karegoudra Jayanth, for their dedicated assistance and constructive feedback. Their expertise was instrumental in refining our methodologies and enhancing the overall quality of our work. Their willingness to engage in thoughtful discussions and offer critical advice has greatly contributed to our learning and the advancement of our project.

Their combined efforts have not only helped us navigate the challenges associated with this project but have also fostered an environment of academic rigor and innovation. We sincerely appreciate their commitment to our growth and their contributions to our academic and professional journey.

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] Ida Momennejad, Hosein Hasanbeig, Felipe Vieira Frujeri, Hiteshi Sharma, Nebojsa Jojic, Hamid Palangi, Robert Ness, and Jonathan Larson. Evaluating cognitive maps and planning in large language models with cogeval. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[16] 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.

7 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 10: GPT-4 reviews on the *Attention is All You Need* paper[16]