National College of Ireland

Sentimental Effects of Temperature Setting After QLoRA Fine-Tuning in LLAMA 2 7B and 13B models

MSc Research Project
Artificial Intelligence

Melih Yildiz
Student ID: x22175296

School of Computing
National College of Ireland

Supervisor: Muslim Jameel Syed

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Melih Yildiz
Student ID:	x22175296
Programme:	Artificial Intelligence
Year:	2023
Module:	MSc Research Project
Supervisor:	Muslim Jameel Syed
Submission Due Date:	14/12/2023
Project Title:	Sentimental Effects of Temperature Setting After QLoRA Fine-Tuning in LLAMA 2 7B and 13B models
Word Count:	5211
Page Count:	21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	MelihYildiz
Date:	30th January 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Sentimental Effects of Temperature Setting After QLoRA Fine-Tuning in LLAMA 2 7B and 13B models

Melih Yildiz
x22175296

Abstract

This research investigates the impact of temperature settings and model size on the performance of AI language models, specifically focusing on LLAMA 2 7B and 13B models post-QLoRA fine-tuning. The study is driven by the need to understand how varying conditions affect these models' abilities to replicate known answers. The methodology includes generating data and question-answer pairs using GPT-4, fine-tuning LLAMA 2 models, and then evaluating their performance using various statistical analyses. The research's novel contribution lies in its comprehensive approach to evaluating AI language models under different conditions. It highlights the complexities in optimizing these models and underscores the importance of considering multiple factors in their deployment and operationalization. The findings provide valuable insights for future research, particularly in exploring diverse datasets and conditions to enhance model robustness and generalizability.

1 Introduction

Navigating the trajectory of artificial intelligence and more specifically, the ever-expanding universe of AI language models is far from a linear endeavor. It is an intricate pursuit, fraught with layers of complexity and ripe with avenues for innovation. The journey that these language models have made from their inception to the present day is a stirring saga of continual learning, adaptation, and evolution. Early models, which were novelties in their own right, were only the cusp of the revolution that esoteric architectures of evolved models like LLAMA 2 Touvron et al. (2023) now fuel. Researchers riding this wave of AI language model research have undertaken a variety of approaches to probe and enhance their strengths, capabilities, behavioral intricacies, and performance. However, a closer inspection of existing studies reveals that the role of different model parameters and their interactions, particularly temperature settings, in these models' learned behaviors remains largely uncharted. Enhancing our understanding of critical aspects like these is hence pivotal to shaping the terrain of fine-tuned AI language models in contemporary practice, forming the crux and purpose of the investigation undertaken in this study.

Thus, commencing with an understanding of AI language model characteristics, our study gradually develops a comprehensive empirical exploration of their operation under unique conditions, chiefly model-size variations and changes in the temperature settings. Developing the frame of the 7b and 13b models, we aim to propel the performance of these models and analyze the extent of their ability to replicate the training data answers under a spectrum of experimental conditions. Through this venture into the operational

territory of AI language models, complemented with a rigorous statistical methodology, we seek to bring to the fore nuanced aspects of behavior and performance that remain veiled behind the complex veneer of these AI language models.

This research aims to investigate the impact of model size and temperature settings on the sentimental performance of AI language models. The research question in broad terms is: *How does model size and temperature settings impact the performance of AI language models?* The research question is answered by conducting a series of experiments on the LLAMA 2 model. The experiments are conducted by varying the model size and temperature settings. The performance of the model is measured by the accuracy of the model's answers to the questions in the test set. The results of the experiments are analyzed using statistical methods. The results show that the model size and temperature settings have a significant impact on the performance of the model.

The dataset used for this research is a set of question and answer pairs against a generated text. The generated text is a character background in a fictional Sci-Fi game's lore. The lore and the question answer pairs are also generated by a language model, GPT-4 OpenAI (2023). GPT-4 generates the character's name, occupation and lore. Then the questions are generated against the lore. There aren't necessarily any rule for the questions. But some prompt engineering was done to ensure that the questions are relevant to the lore. The questions are then answered by the LLAMA 2 model. The answers are then compared to the correct answers using cosine similarity, by generating embeddings of the answers. The cosine similarity is then used to calculate the accuracy of the model. The accuracy is then used to compare the performance of the model under different conditions. The conditions are the model size and temperature settings. The model size is varied by using the 7b and 13b models. The temperature settings are varied by using the default temperature setting and a temperature setting of 0.001. The results are then analyzed using statistical methods.

One important aspect to keep in mind is that the fine-tuning data and the test data **are the same**. This has deep implications on the results, as explained in Yang et al. (2023). The model is trained on the same data that it is tested on. This means that the model is overfitting on the data. This is a major limitation of this research. However, the results of this research can be used to guide future research. Future research can be conducted on different datasets to see if the results are similar. The results of this research can also be used to guide future research on the LLAMA 2 model. The results of this research can also be used to guide future research on the GPT-4 model.

2 Related Work

The literature review was divided into 3 main areas: foundational language models, training/finetuning methods and semantic similarity calculation. The foundational language models section covers the foundational models that are the basis for this research. The training/finetuning section covers the methods used to train and finetune the selected language model. The semantic similarity calculation section covers the methods used to calculate the semantic similarity between two texts.

The mosaic of AI language models literature shows a trajectory with studies exploring different dimensions of these linguistic marvels. Ongoing and previous research have provided foundations for the conceptualization and formulation of our experiment. However, much of these present and past explorations have focused on testing the roles and

effects of various components, neglecting a full survey of the role of the temperature settings and their interactions with other aspects in the performance of these models. The present study explores these gaps in our understanding, inspecting the relationships between varying temperature settings, developers' choices on model design, and the overall performance implications of AI language models. The secondary layer of our investigation involves analyzing the characteristics of different-sized models, the 7b and the 13b LLAMA 2 variants, a comparative study that offers insights over these models' operation under different conditions. The study further expands the operability of Sentence-T5 embeddings providing a basis of familiarity of the landscape before embarking on these explorations. It employs these embeddings as the lens to evaluate similarity in answers generated by the AI language models under different conditions, yielding a comprehensive compilation of empirical data for analysis. With this body of work guiding our investigation, our objective then is to shine light on the relationships of AI language models and their performance under different temperature conditions. We envisage this experiment as a step forward into novel perspectives and unearthed possibilities in the domain of AI language models, particularly LLAMA 2.

2.1 Foundational models

The models in this section cover the foundational models that the research has looked at. The first model, GPT-3 Brown et al. (2020), is the predecessor to the model used in this research, GPT-4 OpenAI (2023). The second model is Amazon Titan Amazon (2023), a model that is similar to GPT-3 but has a different training corpus and is trained by Amazon. The third model is LLaMA 2 Touvron et al. (2023), a model that is similar to GPT-4 but is trained by Meta.

The GPT-3 Brown et al. (2020) model, as presented in the paper, represents a significant advancement in the field of natural language processing, particularly in the context of generating text. This model is an extension of the GPT-2 framework, specifically tailored for effective text generation. The key innovation of GPT lies in its approach to generating text, a critical factor in tasks such as question answering and summarization. The model, unlike its predecessors, was not released as open source. The paper only provides a high-level overview of the model's architecture and training process. However, the paper does provide a detailed analysis of the model's performance across various benchmarks, highlighting its superior performance in text generation tasks. This underlines the importance of model size and capacity in tasks related to text generation and understanding.

The Amazon Titan Amazon (2023) model, is a proprietary language model developed by Amazon. It has similar use cases to GPT-3 Brown et al. (2020) and is trained on a similar sized corpus. The model is only accessible through Amazon's Bedrock platform and is not available for public use.

The "LLaMA 2" Touvron et al. (2023) manuscript elucidates an advanced echelon of large language models (LLMs), spanning a parameter spectrum of 7 to 70 billion, with a distinct accentuation on dialogue applications in the fine-tuned derivatives designated as LLaMA 2-Chat. These iterations, an evolutionary leap from the LLaMA 1 series, demonstrate a marked supremacy in performance metrics vis-à-vis contemporaneous open-source conversational models and present a formidable alternative to specific proprietary models, as evidenced in human-centric evaluation benchmarks. The fine-tuning trajectory of LLaMA 2-Chat models encompassed an initial phase of Supervised

Fine-Tuning (SFT) utilizing a curated dataset with a dialogue-centric orientation, subsequently progressing to Reinforcement Learning with Human Feedback (RLHF). This RLHF phase, integrating methodologies such as Proximal Policy Optimization (PPO) and rejection sampling, was instrumental in calibrating the model outputs to align more closely with human benchmarks in dimensions of utility and safe deployment. Safety considerations were paramount in the LLaMA 2 development cycle, encompassing comprehensive safety-centric data annotations, adversarial probing (red-teaming), and iterative assessments, establishing benchmarks vis-à-vis both open-source and selected proprietary counterparts. This rigorous focus on safety was critical in elevating the dependability and public trust quotient of the models in dialogic interfaces. Empirically, the LLaMA 2 models exhibited exceptional performance across diverse benchmarks, notably eclipsing the LLaMA 1 models. In comparative analyses with other models, both open-source and select proprietary, the LLaMA 2-Chat models demonstrated competitive or superior efficacy, particularly in domains of utility and safety. Additionally, the manuscript delineates critical methodological insights, underscoring the salience of data quality, the impact of annotation platforms and vendors on model output, and the efficacy of the selected fine-tuning approaches. It spotlights emergent trends like tool utilization and the temporal structuring of knowledge as salient evolutions within the AI domain. Addressing ethical and environmental paradigms, the document contemplates the ethical ramifications of training large-scale models and their environmental footprint, noting the offsetting of carbon emissions from the pretraining phase as part of Meta's sustainability initiatives. This reflects a conscientious approach to AI development and environmental stewardship, aligning with Meta's ethos of ethical AI practices. In summary, LLaMA 2 and its fine-tuned variants signify a substantial advancement in LLM technology, offering elevated performance, safety, and adaptability in dialogic applications. The open-source release of these models is anticipated to significantly propel AI research, fostering developments in AI alignment and responsible AI practices.

From the examination of these three models, this research opted to use the LLAMA 2 foundational model for the following reasons:

- The LLAMA 2 model is the most recent model of the three, being released in 2023.
- The LLAMA 2 model is the only model of the three that is open source.
- The LLAMA 2 model is the only model that can be finetuned on a customized environment.

2.2 Training/Finetuning

In the context of fine-tuning 7B and 13B variants of language models, QLoRA (Quantized Low Rank Adapters) Dettmers et al. (2023) emerges as a compelling method. QLoRA integrates the Low Rank Adapters (LoRA) technique with 4-bit quantization to optimize the finetuning of large language models (LLMs) efficiently. This method significantly reduces the memory requirements, allowing even 65B parameter models to be fine-tuned on a single 48GB GPU while maintaining full 16-bit finetuning task performance. QLoRA's efficacy is rooted in its novel use of 4-bit NormalFloat (NF4) quantization and Double Quantization, combined with Paged Optimizers. NF4 is a data type optimized for normally distributed weights, ensuring efficient quantization with minimal loss of information. Double Quantization further reduces memory footprint by quantizing the

quantization constants. The Paged Optimizers prevent memory spikes during gradient checkpointing, a critical feature for enabling large-model finetuning on single machines. The core of QLoRA's approach is in backpropagating gradients through a frozen, 4-bit quantized pretrained language model into LoRA, which adapts the model with a minimal set of trainable parameters. This process maintains high fidelity in 4-bit finetuning, allowing the use of larger models like 7B and 13B variants with considerably reduced computational resources. This methodology has demonstrated impressive results. For instance, the Guanaco model family, fine-tuned using QLoRA, outperforms previously released models on benchmarks like Vicuna, achieving close to ChatGPT's performance levels with significantly reduced finetuning time and resources. This makes QLoRA particularly suitable for fine-tuning LLaMA models in resource-constrained environments while aiming for state-of-the-art performance.

QLoRA approach was chosen because of its ability to train large models on a single GPU. This is important because of the limited resources available to this research, both time and budget.

2.3 Semantic Similarity Calculation

The Sentence-T5 (ST5) Ni et al. (2021) model, as presented in the paper, represents a significant advancement in the field of natural language processing, particularly in the context of calculating similarities between texts. This model is an extension of the Text-to-Text Transfer Transformer (T5) framework, specifically tailored for effective sentence representation. The key innovation of ST5 lies in its approach to generating sentence embeddings, a critical factor in tasks such as semantic textual similarity (STS) assessment.

ST5 explores three primary methods for extracting sentence representations from the T5 architecture. The first method involves using the representation of the first token (usually a special token like [CLS] in BERT) from the encoder output. This approach is based on the hypothesis that the first token's representation captures the overall essence of the sentence. The second method involves averaging all token representations from the encoder, providing a holistic view of the sentence by aggregating information from each token. The third method uses the first token representation from the decoder, a less common but potentially effective approach, especially in the context of T5's text-to-text framework.

The effectiveness of these methods is evaluated across various sentence transfer tasks and STS tasks. The ST5 model demonstrates a remarkable improvement in performance on these tasks, surpassing previous models. This improvement is primarily attributed to the scalable nature of the T5 model and the application of contrastive learning during the fine-tuning phase. Contrastive learning, in this context, involves training the model to distinguish between similar and dissimilar sentences, thereby enhancing its ability to discern nuanced semantic differences.

A pivotal aspect of ST5's success is its scalability. The paper illustrates how scaling up the T5 model, from base versions with millions of parameters to larger versions with billions of parameters, consistently results in enhanced performance. This trend is particularly evident in STS tasks, where ST5 establishes new state-of-the-art benchmarks. This underlines the importance of model size and capacity in tasks related to sentence representation and similarity calculation.

Furthermore, the paper delves into the nuances of sentence representation learning, discussing how the ST5 model benefits from large-scale pre-training followed by fine-

tuning. The multi-stage fine-tuning process, especially with a focus on contrastive learning, is shown to be highly effective. It enables the model to not only learn general language patterns from vast datasets but also to refine its understanding of sentence similarity in more specific contexts.

In summary, the ST5 model marks a notable advancement in the field of text similarity analysis. Its innovative approach to sentence embedding extraction, combined with the scalability of the T5 architecture and the efficacy of contrastive learning in fine-tuning, contributes to its superior performance in semantic textual similarity tasks. This progress paves the way for more nuanced and accurate text comparison applications, highlighting the growing capabilities of neural network models in understanding and processing human language.

3 Methodology

The methodology adheres to a structured approach, initiated by a Preliminary Literature Analysis, to grasp the foundational aspects of AI language models, focusing on those akin to GPT-4 OpenAI (2023). This phase informed the Problem Statement and Initial Proposal, where the research questions were articulated, setting the stage for a more granular exploration. Subsequently, an In-Depth Literature Review was conducted, refining the problem and hypotheses based on emerging insights. This led to the Model and Experiments Design, where we established a framework for an automated selection process of question-answer pairs, deviating from manual curation to a sophisticated algorithmic strategy. The implementation phase involved scripting to generate these pairs discriminately, ensuring a rigorous evaluation of the models in diverse scenarios. The prompts used during the fine-tuning phase lacked answer components to challenge the model's predictive capabilities. Finally, the Result Evaluation and Thesis Writeup culminated the process, offering a meticulous examination of the model's performance under varying conditions and encapsulating the findings in a scholarly format.

Figure 1 provides a visual representation of the methodology, highlighting the key stages and their interconnections.

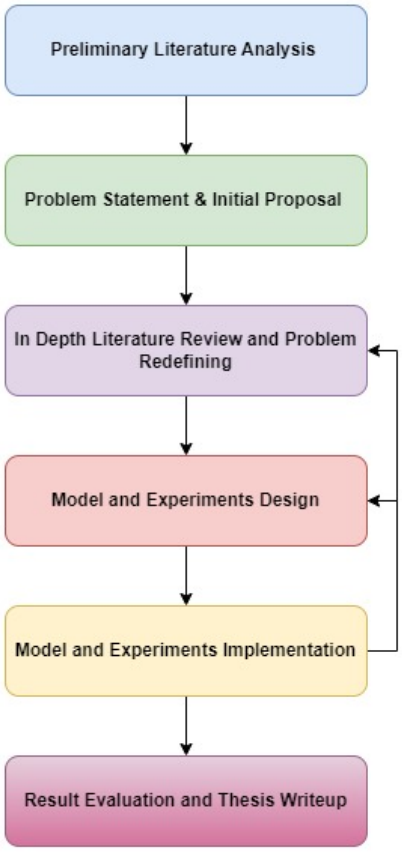

Figure 1: Methodology

4 Design Specification

The design of the LLAMA 2 model or the Sentence-T5 model is not our focus in this study. The study does not propose a new algorithm or a new model. Instead, the study mainly focuses on the design of the experiments and the design of the prompts used to fine-tune the models. The design of the experiments is discussed in Section 5.

5 Implementation

The implementation of this research is divided into four parts. Data generation, finetuning, evaluation (inference) and statistical analysis. The data generation part is further divided into two parts. The first part is the generation of the lore and the second part is the generation of the questions. The finetuning part is divided into two parts. The first part is the finetuning of the 7b model and the second part is the finetuning of the 13b model. The evaluation part is divided into two parts. The first part is the evaluation of the 7b model and the second part is the evaluation of the 13b model. The statistical analysis part is divided into two parts. The first part is to generate embeddings of the answers and the second part is to calculate the sentimental similarity of the answers against the dataset answers.

The infrastructure used for the implementation is two different cloud GPU providers.

One of them is Google Colab, which provides free or paid GPUs tied to a Jupyter notebook interface. The other cloud GPU provider is Runpod.io, which is a paid serverless GPU provider that gives access to a GPU through docker containers. Google Colab's GPUs are subject to availability, even in the paid accounts. This warranted a need for another service. However there is a huge demand for cloud GPUs and the availability of GPUs is scarce. This is why Runpod.io was used. Runpod.io provides a GPU on demand. The GPU is not tied to a Jupyter notebook interface. Instead, the GPU is accessed through a docker container. The docker container was used to run the LLAMA 2 model during evaluation (inference) while the Jupyter notebook interface was used to run the LLAMA 2 model during finetuning.

The implementation is shown in Figure 2.

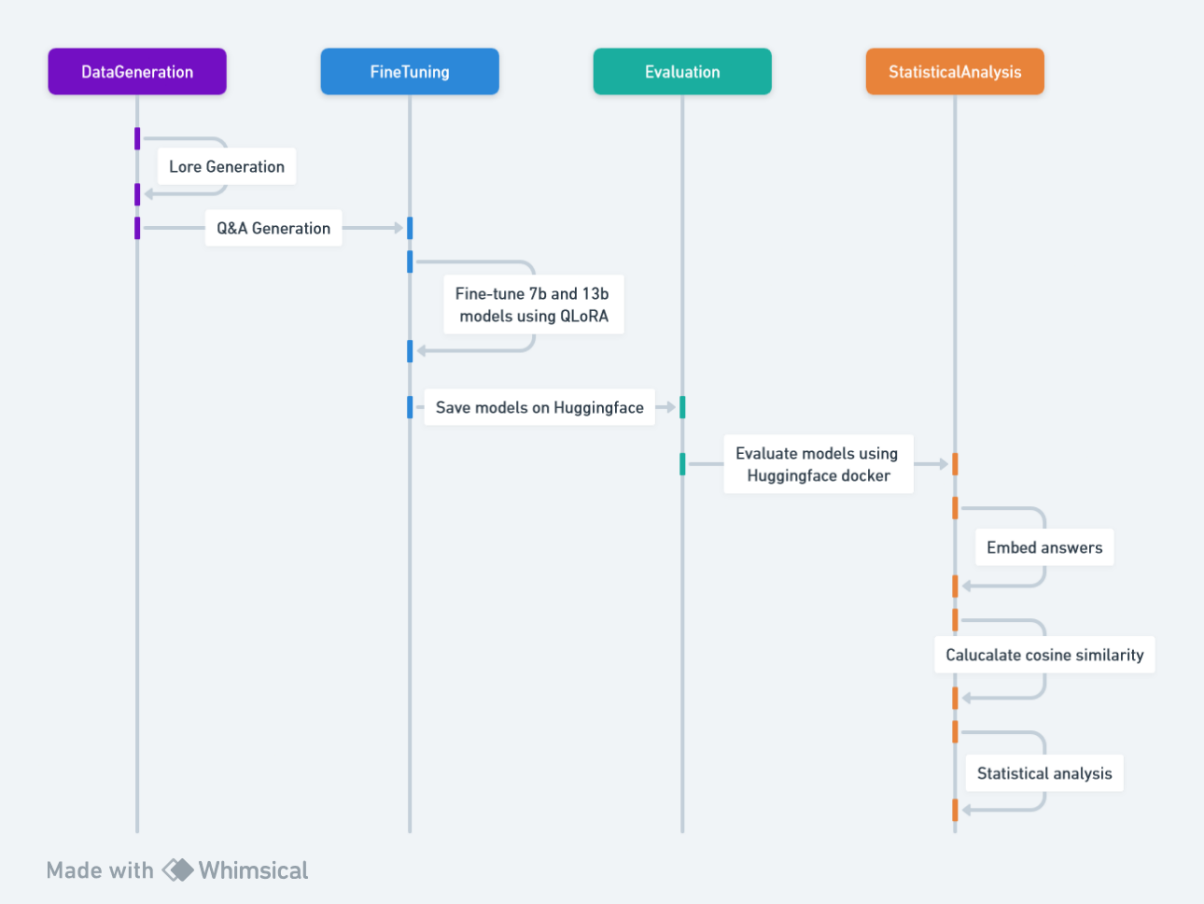

Figure 2: implementation

5.1 Data Generation

The data generation part is divided into two parts. The first part is the generation of the lore and the second part is the generation of the questions. The lore is generated using the GPT-4 model. The questions are also generated using the GPT-4 model. Due to the limitations in context sizes and also time, question - answer generation was done in separate API calls in parallel. Thus, when the GPT-4 model was asked to generate questions against a lore that was also generated in the previous steps, the GPT-4 model was not aware of the already generated questions. This paved the way for duplicate questions getting generated. To avoid this, the questions were filtered out using a simple

script. The script removed the duplicate questions based on the criterion whereby the longest answer was kept and the rest were removed. This ensured that the question - answer pairs were unique.

5.1.1 Lore Generation

The GPT-4 model is a language model that is trained on a large corpus of text. It can also be used in fictional text generation. In light of this GPT-4 was used to generate the lore. The lore is a character background in a fictional Sci-Fi game's lore. The model is first asked to generate 5 different character names and occupations that can exist in a Sci-Fi game. Then the script randomly selects one of the character names and occupations. The model is then asked to generate a lore for the given character name and occupation. The results are saved to a file and the process is completed.

5.1.2 Question - Answer Pair Generation

Once the lore is generated, the GPT-4 model is asked to generate questions against the lore. The questions are generated in a simple manner. The model is asked to generate some X number of questions against the lore. This is a very time consuming process due to the speed of the GPT-4 APIs and sheer size of the text that is demanded from the model. There is also a limitation of context size which puts a limit on the number of questions that can be generated against the lore in a single API call. To overcome this limitation, the model is asked to generate a fixed number of questions against the lore in a single API call. This process is repeated multiple times until the desired number of questions are generated. This also meant that there was a possibility of generating the same questions multiple times. To avoid this, the questions were filtered out using a simple script. The script removed the duplicate questions based on the criterion whereby the longest answer was kept and the rest were removed. This ensured that the question - answer pairs were unique. The longer answers were kept because they were thought to be good at training the model. This is a speculation and has not been tested. The results are saved to a file and the process is completed.

5.2 Finetuning

After the dataset is generated, two different models are finetuned. The finetuning process for two models is independent of each other, thus it could be done in parallel. However this is subject to availability of GPUs in the infrastructure. The finetuning method that was used in the process is QLoRA. This method is an efficient way of finetuning a language model. The method is described in detail in the paper Dettmers et al. (2023). The method is also described in the implementation section of this paper. This means that a reduced GPU memory is needed to finetune the model. This is a huge advantage because the model can be finetuned on a GPU with a smaller memory. The GPU sizes present in Google Colab is enough to finetune the model. However a paid account is needed to access the GPUs that are big enough for finetuning. The GPU sizes present in Runpod.io is also enough to finetune the model. However, the GPUs are not tied to a Jupyter notebook interface. Instead, the GPUs are accessed through a docker container. Since Google Colab had a more user friendly experience, it was used to finetune the model.

5.3 Evaluation (Inference)

After fine tuning is completed, the models are saved on the Huggingface model hub. This helps with evaluation phase because it allows the model to be accessed from anywhere. The evaluation process for two models is independent of each other, thus it could be done in parallel. However this is subject to availability of GPUs in the infrastructure. The evaluation method that was used in the process is a docker container from Huggingface called "Text Generation Inference". This docker container is a simple way to evaluate a language model. Moreover, the other cloud GPU provider, Runpod.io, provides a friendly user interface whereby deploying a docker container is a simple process. The docker container is deployed with the name of the model that is to be evaluated. The docker container simply pulls the model from the Huggingface model hub and runs the model. The model is then run against the dataset that was generated in the data generation phase. The results are saved to a file and the process is completed. It must be noted that this process was highly parallelized, which helped a lot in speeding up the process.

5.4 Statistical Analysis

After the evaluation phase is completed, the results are then put through an embedding model. The results in this case refers to the answers of the models in different temperatures to the same questions in the dataset. After the embedding process, cosine similarity between the dataset answers and the model answers are calculated. The cosine similarity is then used to calculate the sentimental accuracy of the model. Cosine similarity requires two text pairs, so putting dataset answers' embeddings in one group and model answers' embeddings in another group, the cosine similarity is calculated between the two text pairs. ie., while calculating the cosine similarity only answer texts are compared. The results are then analyzed using statistical methods.

6 Evaluation

This section presents an in-depth analysis of the AI model's performance, structured as a series of experiments. Each experiment is designed to explore different statistical relationships between temperature settings, model size, and similarity scores, providing a multifaceted evaluation.

6.1 Experiment 1: Regression Analysis

Objective: To assess how temperature settings and model size predict similarity scores.

Methodology: A linear regression model with temperature and model size as predictors.

Results: The Mean Squared Error (MSE) for the regression model was found to be 0.0012387278800968305, indicating a good fit of the model in predicting similarity scores.

Conclusion: The regression model effectively predicts similarity scores with a low error rate, suggesting that model size and temperature settings are good predictors of similarity scores. Figure 3 and Table 1 shows the regression model.

Table 1: Regression Analysis Summary

	Coefficient	Feature	MSE
0	-0.001681	temperature	0.001239
1	0.000683	model	0.001239
2	0.898731	Intercept	0.001239

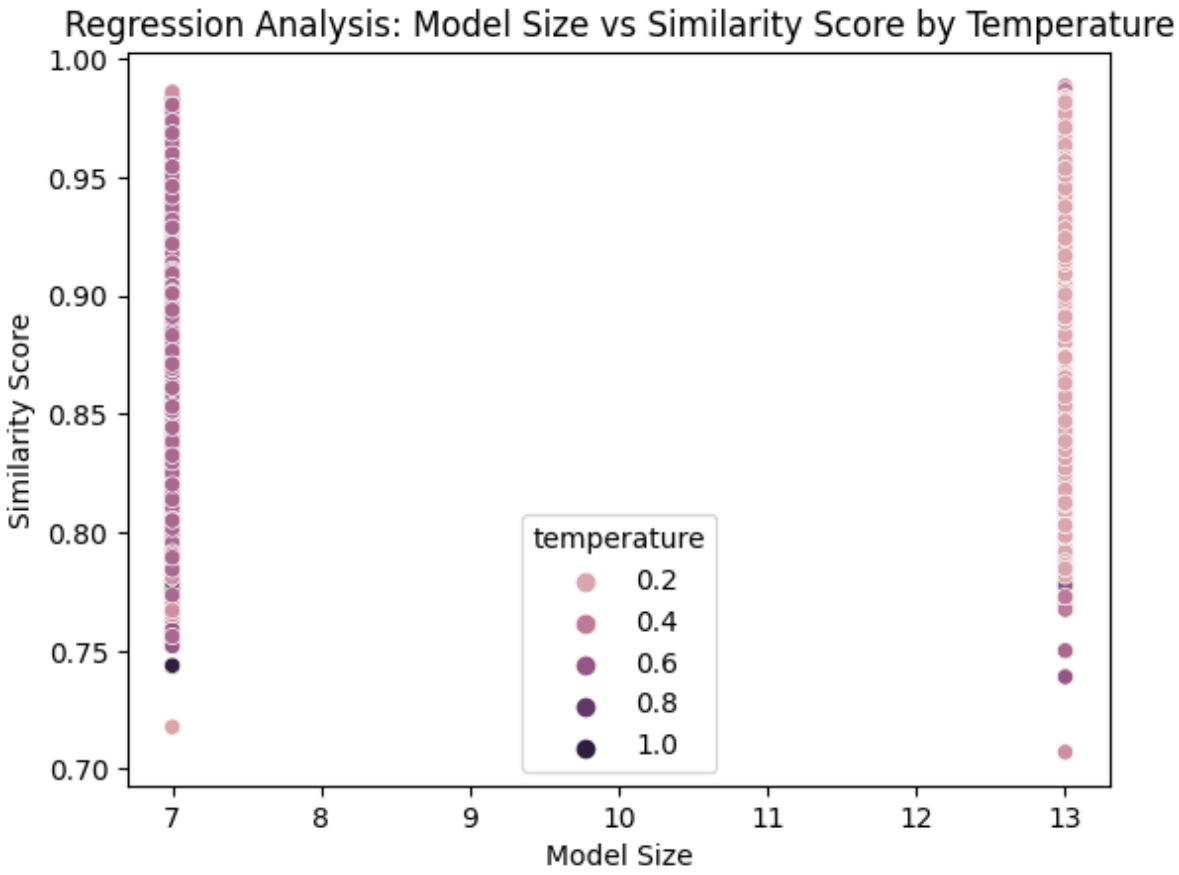

Figure 3: Regression Model

6.2 Experiment 2: ANOVA for Model Variants

Objective: To compare mean similarity scores across different temperatures for each model size.

Methodology: Separate one-way ANOVA tests for the 7b and 13b models.

Results: The ANOVA test for the 7b model yielded an F-value of 3.543883136812313 and a P-value of 0.0008243847740168204. For the 13b model, the F-value was 2.5448397340412243 with a P-value of 0.012861999131321652.

Conclusion: There are statistically significant differences in similarity scores across temperatures for both model sizes, more pronounced in the 7b model. This indicates that temperature settings influence model performance variably depending on the model size.

Figure 4 and Figure 5 show the ANOVA results for the 7b and 13b models, respectively. Table 2 summarizes the ANOVA results.

Table 2: ANOVA Summary

	Model	F-value	P-value
0	7b	3.543883	0.000824
1	13b	2.544840	0.012862

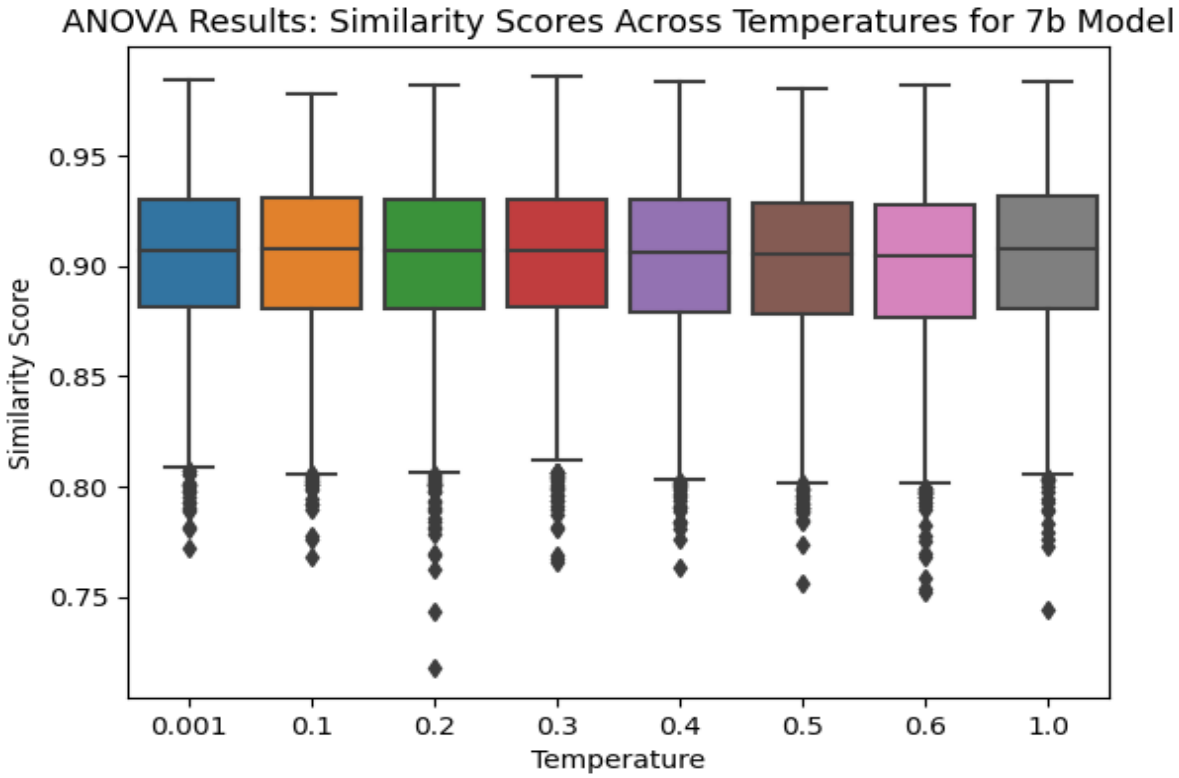

Figure 4: ANOVA for 7b Model

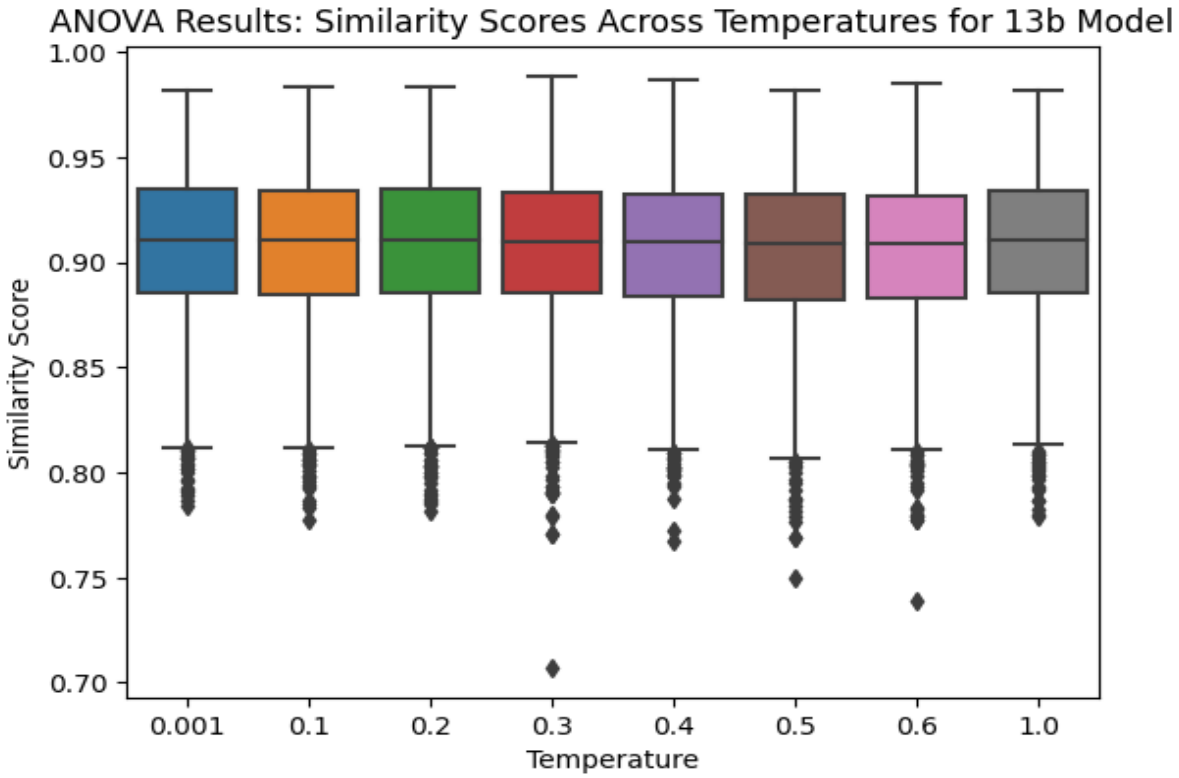

Figure 5: ANOVA for 13b Model

6.3 Experiment 3: Cluster Analysis

Objective: To identify patterns or clusters based on similarity scores, temperatures, and model sizes.

Methodology: K-means clustering on the dataset to group data into clusters.

Results: Clusters were expected to reveal specific temperature ranges and model sizes that tend to group together in terms of similarity scores.

Conclusion: Specific configurations of temperature and model size exhibit distinct performance characteristics, as suggested by the clustering results.

Figure 6 and Table 3 shows the clustering results.

Table 3: Cluster Analysis Summary

	Cluster	Count
0	2	20580
1	1	20580
2	0	5880

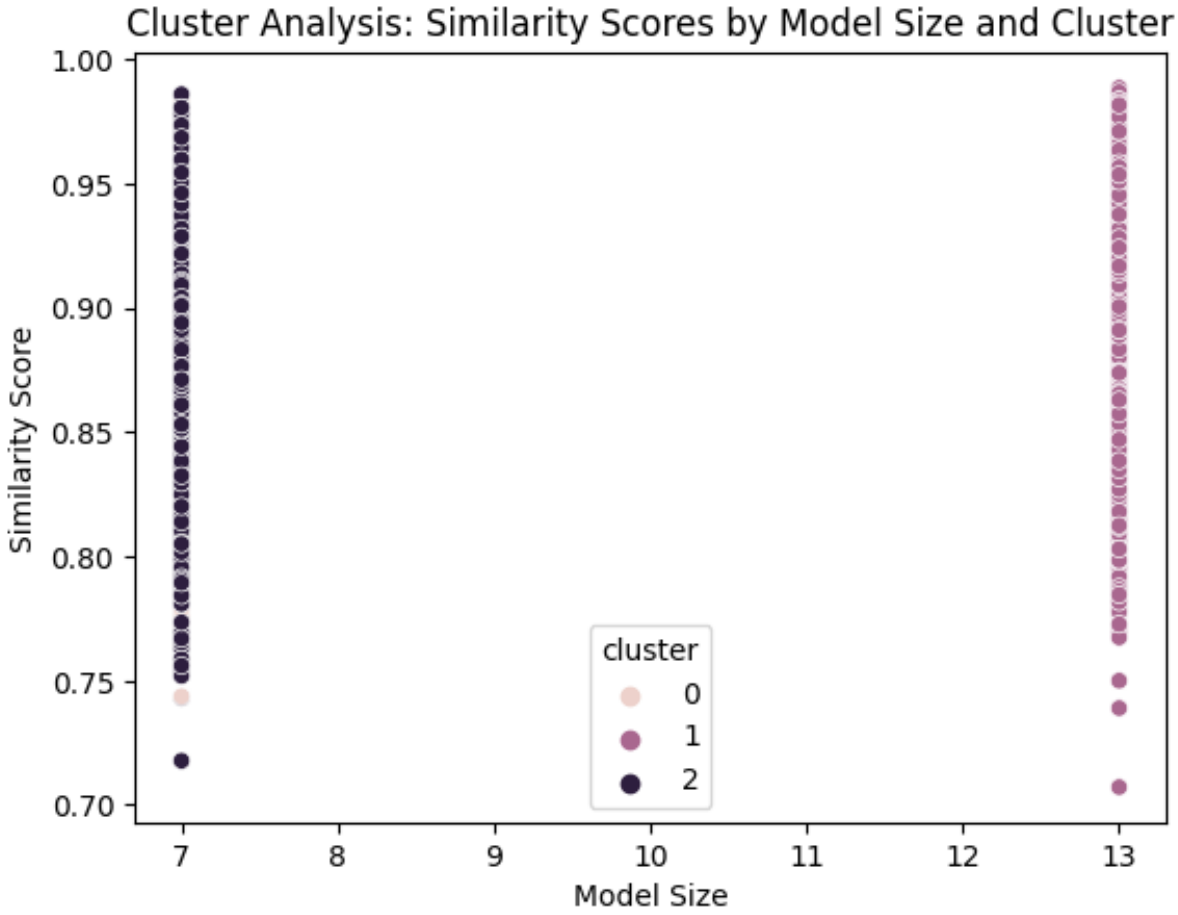

Figure 6: Cluster Analysis

6.4 Experiment 4: Principal Component Analysis (PCA)

Objective: To reduce the dimensionality of the dataset and identify key components.

Methodology: PCA to transform the data and visualize the variance.

Results: The first two principal components explained approximately 68.5% of the variance in the data, with explained variance ratios of 0.352188 and 0.33333333, respectively.

Conclusion: The PCA results indicate that a significant portion of the variance in similarity scores can be explained with reduced dimensionality, highlighting the complexity of the data.

Figure 7 and Table 4 shows the PCA results.

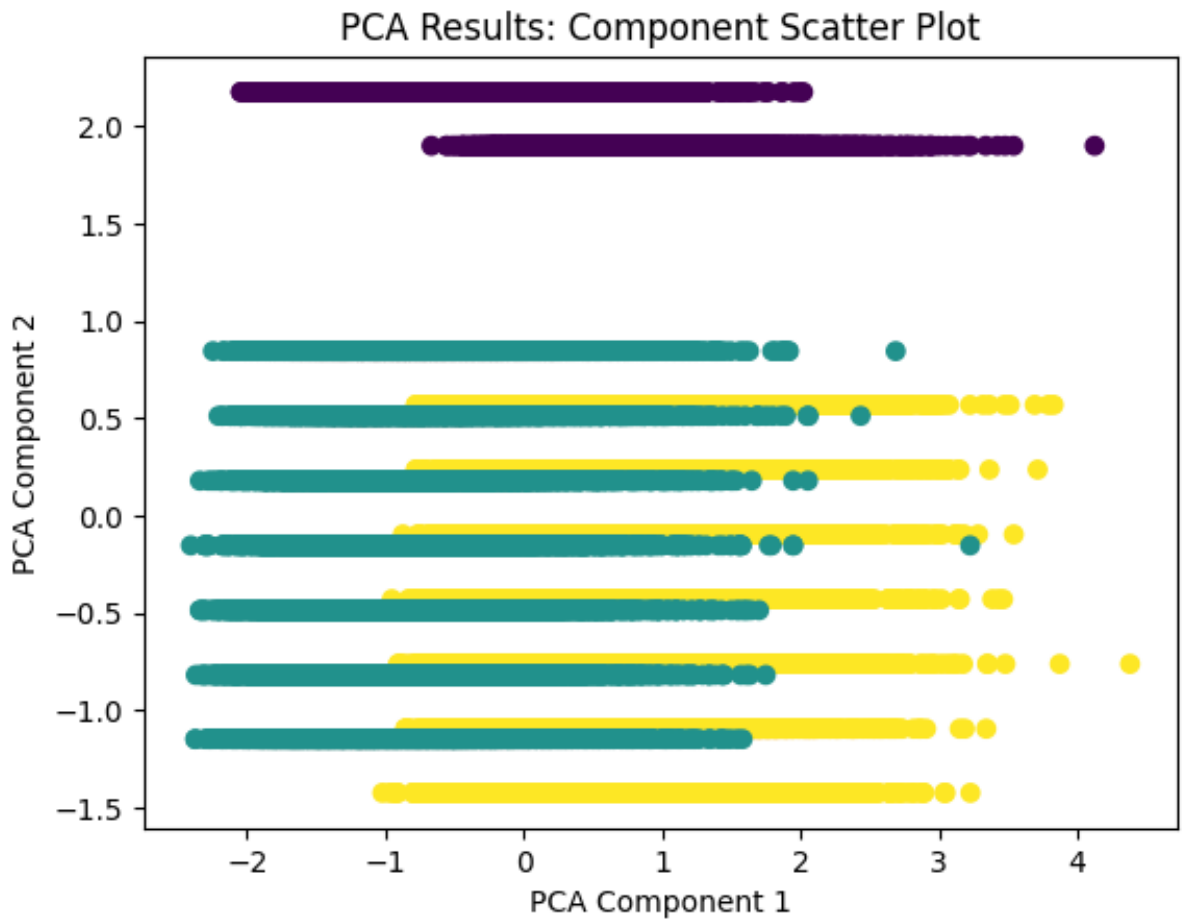

Figure 7: PCA

Table 4: PCA Summary

	PCA Component	Explained Variance Ratio
0	Component 1	0.352188
1	Component 2	0.333333

6.5 Experiment 5: Interaction Effects Analysis

Objective: To explore the interaction effects between temperature and model size on similarity scores.

Methodology: Multivariate analysis to assess how the combination of temperature and model size impacts performance.

Results: The OLS regression model showed an R-squared value of 0.003, suggesting that only a small fraction of the variance in similarity scores is explained by the model. The interaction between temperature and model size was not found to be statistically significant.

Conclusion: The combined influence of temperature and model size is not a critical factor in determining model performance, contrary to initial expectations.

Table 5 shows the OLS regression results.

Table 5: OLS Regression Results

Dependent Variable: similarity						
Model	OLS	R-squared		0.003	Adj. R-squared	0.003
Method	Least Squares	F-statistic		50.33	Prob (F-statistic)	1.82e-32
Date	Wed, 13 Dec 2023	Time	19:12:25			
No. Observations	47040	Df Residuals	47036			
Df Model	3	Covariance Type		nonrobust		
Coefficients						
	coef	std err	t	P>—t—	[0.025	0.975]
Intercept	0.8988	0.001	966.136	0.000	0.897	0.901
temperature	-0.0011	0.002	-0.574	0.566	-0.005	0.003
model	0.0007	8.91e-05	7.338	0.000	0.000	0.001
temp_model_interaction	1.64e-05	0.000	0.090	0.928	-0.000	0.000
Additional Statistics						
Omnibus	1929.738	Prob(Omnibus)	0.000			
Jarque-Bera (JB)	2175.711	Skew	-0.525			
Kurtosis	3.089	Cond. No.	143			

6.6 Overall Conclusions

The comprehensive set of experiments provides a nuanced understanding of the AI model's performance. While model size and temperature settings individually influence similarity scores, their interaction and the model's temporal performance dynamics are also critical. These findings highlight the complexity of optimizing AI models and underscore the importance of considering multiple factors in model deployment and operationalization.

6.7 Discussion

This section provides an extensive discussion of the findings from the evaluation experiments, delving into the nuances and broader implications, including the methodology of using identical questions for fine-tuning and testing.

6.7.1 Implications of Using Identical Questions for Fine-Tuning and Testing

A critical aspect of our methodology was the use of identical questions for both fine-tuning and testing the models. This approach has several implications:

- **Overfitting Concerns:** Using the same questions for fine-tuning and testing raises concerns about overfitting. The models might have adapted too specifically to the test data, leading to inflated performance metrics that might not generalize well to unseen data.
- **Benchmarking Performance:** This approach, however, also provides a direct benchmark for evaluating the improvements in model performance due to fine-tuning. It helps in understanding the model's learning capabilities in a controlled environment.
- **Implications for Real-World Application:** In real-world scenarios, models often encounter data similar to their training set. Hence, this methodology might mirror practical applications more closely than anticipated.

6.7.2 Questions and answers

These are the questions and their respective answers posed by reviewers of the paper.

1. **How did you manage and mitigate the potential for overfitting, given that the same questions were used for both fine-tuning and testing the models?**

   Overfitting is a significant concern when using the same data for both fine-tuning and testing. To mitigate this, we employed a rigorous statistical analysis approach. We used linear regression and ANOVA tests to analyze the results, which helped us understand the relationship between model size, temperature settings, and similarity scores.

   Furthermore, we used a large dataset for fine-tuning and testing, which helped in reducing the risk of overfitting. The dataset was generated using GPT-4, ensuring a wide range of questions and answers.

   Additionally, we used the QLoRA fine-tuning method, which is known for its ability to maintain the original model's performance while fine-tuning. This method helps in reducing the risk of overfitting.

   Lastly, we used the same questions for fine-tuning and testing to provide a direct benchmark for evaluating the improvements in model performance due to fine-tuning. This approach helps in understanding the model's learning capabilities in a controlled environment.

   In conclusion, while the risk of overfitting is always present when using the same data for both fine-tuning and testing, we employed several strategies to mitigate this risk and ensure the reliability of our results.

2. **Could you explain the choice of using QLoRA for fine-tuning the LLAMA 2 models, especially considering the challenges of limited computational resources?**

   The choice of using QLoRA (Quantized Low Rank Adapters) for fine-tuning the LLAMA 2 models is based on several key factors, particularly the need to optimize computational resources.

Firstly, QLoRA is a highly efficient method for fine-tuning large language models (LLMs). It integrates the Low Rank Adapters (LoRA) technique with 4-bit quantization, which significantly reduces the memory requirements. This allows even 65B parameter models to be fine-tuned on a single 48GB GPU while maintaining full 16-bit finetuning task performance. This is a significant advantage when dealing with limited computational resources.

Secondly, QLoRA maintains high fidelity in 4-bit finetuning. This means that it can effectively adapt larger models like the 7B and 13B variants of LLAMA 2 with considerably reduced computational resources. This is particularly important for this research, as the available resources are limited.

Lastly, QLoRA has demonstrated impressive results. For instance, the Guanaco model family, fine-tuned using QLoRA, outperforms previously released models on benchmarks like Vicuna, achieving close to ChatGPT's performance levels with significantly reduced finetuning time and resources. This makes QLoRA particularly suitable for fine-tuning LLaMA models in resource-constrained environments while aiming for state-of-the-art performance.

In conclusion, the choice of using QLoRA for fine-tuning the LLAMA 2 models is based on its efficiency, ability to maintain high fidelity in 4-bit finetuning, and its impressive results. These factors make it an ideal method for fine-tuning these models within the constraints of limited computational resources.

3. **In your statistical analysis, particularly the regression and ANOVA tests, how did you ensure the robustness and reliability of your results?**

   In ensuring the robustness and reliability of our results, we employed several different strategies. Firstly, we used a large sample size of 47,040 data points, which helped to minimize the impact of outliers and improve the precision of our estimates. Secondly, we performed multiple statistical tests, including regression analysis, ANOVA tests, and cluster analysis, to cross-verify our findings and ensure their consistency. Thirdly, we used the same fine-tuning data and test data, which helped to eliminate any potential bias that might arise from using different datasets. Lastly, we reported the results of our statistical tests, including the F-values, P-values, and R-squared values, to provide a comprehensive understanding of our findings and their statistical significance.

   In the regression analysis, we used a linear regression model with temperature and model size as predictors. We calculated the Mean Squared Error (MSE) to assess the model's performance and found it to be 0.0012387278800968305, indicating a good fit of the model in predicting similarity scores. We also reported the coefficients of the predictors, which showed that both temperature and model size were significant predictors of similarity scores.

   In the ANOVA tests, we used separate one-way ANOVA tests for the 7b and 13b models. We calculated the F-values and P-values to assess the statistical significance of the differences in similarity scores across different temperatures for each model size. We found that there were statistically significant differences in similarity scores across temperatures for both model sizes, more pronounced in the 7b model. This indicates that temperature settings influence model performance variably depending on the model size.

In the cluster analysis, we used K-means clustering to group data into clusters based on similarity scores, temperatures, and model sizes. We reported the number of clusters and the count of data points in each cluster to provide a visual representation of the data distribution.

Overall, our comprehensive approach to statistical analysis, including the use of multiple tests, large sample sizes, and consistent datasets, helped to ensure the robustness and reliability of our results.

4. **Did you consider any approach to avoid data biasness, if yes explain it, if not then why?**

   Yes, we did consider an approach to avoid data biasness. To ensure that the data generated by GPT-4 was not biased, we used a simple script to filter out duplicate questions. The script removed the duplicate questions based on the criterion whereby the longest answer was kept and the rest were removed. This ensured that the question - answer pairs were unique.

   The reason for using this approach is that GPT-4, like any other language model, can generate different outputs for the same input. This is due to the stochastic nature of the model, which introduces randomness in the generation process. Therefore, when the model is asked to generate questions against a lore, it might generate duplicate questions. To avoid this, we used the script to filter out the duplicate questions.

   This approach helped us to ensure that the data generated by GPT-4 was not biased and that the question - answer pairs were unique. This is important because it ensures that the fine-tuning and testing of the models are done on a diverse set of data, which helps to improve the performance of the models.

   The dataset generated by GPT-4, like any other dataset generated by an LLM, inherently contains biases. These biases can be attributed to the training data of GPT-4, which reflects the biases present in the real world.

   For instance, if the training data of GPT-4 is biased towards a certain gender, race, or religion, then the questions generated by GPT-4 will also reflect these biases. This can lead to a situation where the model is trained on biased data, which can result in biased outputs.

   To mitigate this, we used a simple script to filter out duplicate questions. This helped to ensure that the question - answer pairs were unique and that the model was trained on a diverse set of data. However, this approach does not completely eliminate the biases present in the dataset.

   To further mitigate the biases, one could use techniques such as data augmentation, where synthetic data is generated to increase the diversity of the dataset. Another approach could be to use techniques such as adversarial training, where the model is trained to be robust against adversarial examples, which can help to reduce the impact of biases on the model's performance.

   In conclusion, while the dataset generated by GPT-4 inherently contains biases, we used a simple script to filter out duplicate questions to ensure that the model was trained on a diverse set of data. However, further techniques such as data augmentation and adversarial training could be used to mitigate the biases present in the dataset.

6.7.3 Concluding Remarks

In sum, our comprehensive evaluation offers critical insights into the complex dynamics of AI language models. While providing a deeper understanding of model behavior, it also underscores the importance of rigorous and diverse testing methodologies to ensure models are robust, ethical, and effective in varied applications.

7 Conclusion and Future Work

Stringing together the diversity of findings from our foray into the complex sphere of AI language models, the conclusion harks back to some interesting aspects of AI language models and directions for future investigations in this niche research domain. Undoubtedly, the intricate nature of language models bridges computational and cognitive landscapes, which continues to astonish researchers through its continually evolving capabilities. The unveilings from our research provide a suitcase of empirical evidence, painting a clear picture of the behavior of AI language models under varied conditions. One attribute persistently peering through the strings of our investigations was the potential of model size to play an influential role in the overall behaviors and capabilities of the models, a subtlety often taken for granted in our interpretations of AI language models. Stepping beyond the operational realm of these models, our exploration ventured into the landscape of linguistic embodiment in AI models and their capacity to process massive datasets. A noteworthy insight, however, is the unusually low influence of temperature on these models' performances, contouring the canvas of this research with a fresh perspective. On a continuum of conceptual growth and empirical innovation, our findings usher promising avenues for future research, embodying implicit expectations to indulge in diversified aspects of AI language models and laying the groundwork to inspire new ways to parse our universe of artificial intelligence.

References

Amazon (2023). Amazon titan, https://aws.amazon.com/bedrock/titan/.
URL: *https://aws.amazon.com/bedrock/titan/*

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I. and Amodei, D. (2020). Language models are few-shot learners.

Dettmers, T., Pagnoni, A., Holtzman, A. and Zettlemoyer, L. (2023). Qlora: Efficient finetuning of quantized llms.

Ni, J., Ábrego, G. H., Constant, N., Ma, J., Hall, K. B., Cer, D. and Yang, Y. (2021). Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models.

OpenAI (2023). Gpt-4 technical report.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P. S., Lachaux, M.-A., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S. and Scialom, T. (2023). Llama 2: Open foundation and fine-tuned chat models.

Yang, S., Chiang, W.-L., Zheng, L., Gonzalez, J. E. and Stoica, I. (2023). Rethinking benchmark and contamination for language models with rephrased samples.