A SYSTEMATIC STUDY OF THE ROLE OF DATA QUALITY AND ALIGNMENT FOR FINE-TUNING LLMS FOR ENHANCED AUTOFORMALIZATION

Krrish Chawla, Mario DePavia, Aryan Sahai, Brando Miranda *
Department of Computer Science
Stanford University
Stanford, CA 94305, USA
{krrish,mariodp,aryan22}@cs.stanford.edu

ABSTRACT

This study explores the role of data quality, particularly alignment, in fine-tuning Large Language Models (LLMs) for the task of autoformalization. Contrary to the conventional emphasis on dataset size, our research highlights the importance of data alignment - the similarity between training data and target domain. Through our experiments, we demonstrate a negative correlation between data alignment and model perplexity loss. These findings suggest a re-evaluation of LLM training approaches, emphasizing quality and relevance over quantity, especially in specialized applications such as autoformalization.

1 INTRODUCTION

Traditional approaches in LLM development have predominantly focused on the scale of datasets used in fine-tuning, a perspective extensively discussed in (OpenAI, 2023). Our research diverges from this norm, proposing that augmented data alignment not only improves LLM efficacy but also establishes a measurable standard for data quality. We demonstrate that enhanced data alignment is associated with a decrease in perplexity loss, which directly contributes to the improved performance of LLMs. This introduction lays the groundwork for our in-depth analysis, further elaborated in the appendix (B). A key focus of our study is the application of these findings in the realm of autoformalization—the conversion of text from natural language to formal programming languages, a topic explored in (Wu et al., 2022).

2 METHODS

In our experiment, we include both code and proof-based datasets: using the Docstring dataset as the benchmark for the former and the AF dataset for the latter. More on the datasets in appendix F. Alignment is a rigorous metric that has a large impact on the effectiveness of a dataset in training LLMs to perform autoformalization accurately. To measure alignment, we use the Task2Vec Alignment Coefficient, which is calculated by the following equation (Lee et al., 2023):

$$\hat{\text{ali}}\text{gn}(D_1, D_2) = 1 - \mathbb{E}_{B_1 \sim D_1, B_2 \sim D_2} d(\hat{f}_{B_1}, \hat{f}_{B_2})$$

Perplexity indicates the LLM's performance in autoformalization, with lower scores representing better performance. For a detailed explanation of perplexity calculation, please refer to the appendix, D. For information on how this directly tests our hypothesis, please refer to the appendix, C.

Finally, we also conducted some experiments to further back up our claim and clarify the reason for some of our assumptions. These can be found in the appendix, K.

*Use footnote for providing further information about author (webpage, alternative address)—*not* for acknowledging funding agencies. Funding acknowledgements go at the end of the paper.

3 EXPERIMENTATION

In our experimentation process, we employed GPT-2 (Radford et al., 2019) as the base model for fine-tuning. We used the standard Huggingface Transformers script (Wolf et al., 2020) to fine tune for language modelling. GPT-2, known for its robust performance in a wide range of NLP tasks, provided a solid foundation for our exploration into the effectiveness of data alignment in autoformalization. The experiments were conducted with a batch size of 8. This size was selected to balance the computational efficiency and the learning capability of the model across different datasets. We fine-tuned the model for a total of 3 epochs. This duration was chosen to ensure sufficient model exposure to the training data while mitigating the risk of overfitting, particularly given our focus on assessing the model's performance with regard to data alignment.

The aim of our experiments was to scrutinize the relationship between data alignment and model performance, as measured by perplexity loss on the test set. We maintained a consistent number of tokens (approx. 4000) for each dataset to fine tune, as well as for the test set to ensure uniformity in data quantity while varying data alignment. The number of examples selected was based on the total number of tokens being in the limit, within margin of error. This controlled setup allowed us to isolate the effect of alignment on model performance.

To further validate our findings and ensure reproducibility, we included detailed information on the preprocessing steps in appendix E and dataset specifics in appendix F.

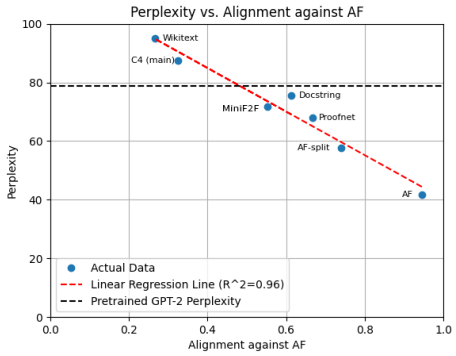

Figure 1: Proof Dataset Results: Alignment scores (table in appendix G) plotted against PPL (table in appendix H) suggests a negative linear correlation and mirrors our expected findings

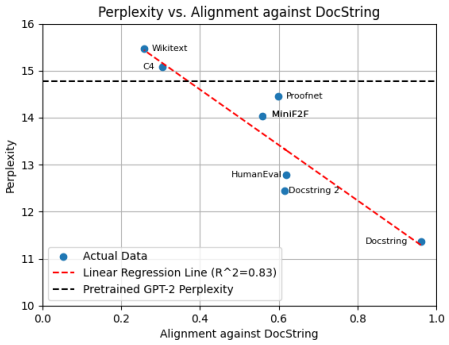

Figure 2: Code Dataset Results: Alignment scores (table in appendix I) plotted against PPL (table in appendix J) suggests a negative linear correlation and mirrors our expected findings

4 DISCUSSION

Our results significantly support the thesis that higher data alignment in training Large Language Models (LLMs) leads to improved performance. This was evident as the LLM fine-tuned on the highly aligned AF dataset showed superior performance with the lowest perplexity score, compared to other datasets like Proofnet and C4, which had lower alignment scores and higher perplexity. This pattern was consistent across both math proof and code datasets, demonstrating a robust negative correlation between data alignment and model perplexity, as indicated by a high r^2 value of 0.96 for math proof datasets and 0.83 for code datasets. These findings suggest that aligning training data closely with the target domain can significantly enhance model performance.

5 LIMITATIONS AND FUTURE WORK

Our study, while revealing significant insights into data alignment in LLMs, faces certain limitations. The primary constraints include reliance on subsets of larger datasets, which may not fully represent their complete characteristics, and the potential impact of unexplored factors like internal

dataset diversity on model performance. Moreover, our computational resources limited the scope of training, affecting the scalability of our findings. Future research should address these limitations by incorporating more diverse and comprehensive datasets, and by expanding computational capacity. We have discussed a few limitations in appendix L and conducted a few experiments to rule out the subset issue in appendix K.

6 URM STATEMENT

The authors acknowledge that all authors of this work meet the URM criteria of ICLR 2024 Tiny Papers Track.

REFERENCES

Zhangir Azerbayev, Bartosz Piotrowski, Hailey Schoelkopf, Edward W. Ayers, Dragomir Radev, and Jeremy Avigad. Proofnet: Autoformalizing and formally proving undergraduate-level mathematics, 2023.

Shai Ben-David, Teresa Luu, Tyler Lu, and D avid P al. Impossibility theorems for domain adaptation, 2023.

Calum Bird. The stack smol python docstrings, 2023a.

Calum Bird. The stack dedup python docstrings 1.0 percent unified, 2023b.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

Tatsunori Hashimoto. Model performance scaling with multiple data sources, 2021.

Alycia Lee, Brando Miranda, Sudharsan Sundar, and Sanmi Koyejo. Beyond scale: the diversity coefficient as a data quality metric demonstrates llms are pre-trained on formally diverse data, 2023.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016.

Brando Miranda. Ultimate utils - the ultimate utils library for machine learning and artificial intelligence, 2021.

Niklas Muennighoff, Qian Liu, Armel Zebaze, Qinkai Zheng, Binyuan Hui, Terry Yue Zhuo, Swayam Singh, Xiangru Tang, Leandro von Werra, and Shayne Longpre. Octopack: Instruction tuning code large language models. *arXiv preprint arXiv:2308.07124*, 2023.

OpenAI. Gpt-4 technical report, 2023.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners, 2019. Accessed: [insert date of access].

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv e-prints*, 2019.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Perric Cistac, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, 10 2020. Association for Computational Linguistics.

Yuhuai Wu, Albert Q. Jiang, Wenda Li, Markus N. Rabe, Charles Staats, Mateja Jamnik, and Christian Szegedy. Autoformalization with large language models, 2022.

Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. Minif2f: a cross-system benchmark for formal olympiad-level mathematics. *arXiv preprint arXiv:2109.00110*, 2021.

A APPENDIX

B RELATED WORK

Our research, focusing on the enhancement of autoformalization through improved data alignment in LLM fine tuning sets, builds upon a select body of work that specifically addresses the nuances of data quality in the context of LLM training.

The concept of data alignment, central to our study, is informed by the findings of "Beyond Scale: The Diversity Coefficient as a Data Quality Metric Demonstrates LLMs are Pre-trained on Formally Diverse Data" by (Lee et al., 2023). This work demonstrated the use of a diversity coefficient derived from the Fisher Information Matrix as a measure of data quality. We extend the discussed importance of data quality to include alignment and its impacts LLM performance, particularly in the domain of autoformalization.

Additionally, "Language Models are Few-Shot Learners" by (Brown et al., 2020) challenges conventional NLP approaches by emphasizing the capacity of scale-driven LLMs in few-shot learning settings. Our research aims to accomplish a similar goal by demonstrating that the fine-tuning process can be significantly shortened when the LLM is fine-tuned with data that is closely aligned with the target domain.

"Impossibility Theorems for Domain Adaptation" by (Ben-David et al., 2023) discusses the significance of similarity between training and test distributions in domain adaptation through formal proofs. We draw a parallel to this in our experimental study by using a quantified measure of data alignment to demonstrate that higher alignment corresponds to improved performance in LLMs, highlighting the importance of relevant and domain-specific training data.

Furthermore, "Model Performance Scaling with Multiple Data Sources" by (Hashimoto, 2021) explores the impact of different methods of combining data to form a training set from various sources on model performance. This resonates with our investigation into how data alignment affects the efficiency of LLMs trained for specific tasks like autoformalization, underlining the importance of not just data quantity, but its contextual relevance and alignment.

Finally, the paradigm shift introduced by "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" by (Devlin et al., 2019), particularly the focus on contextually rich, bidirectional pre-training, emphasizes the importance of factors other than scale which aligns with our hypothesis that the quality of data, including its alignment with the task at hand, is crucial for effective fine-tuning of LLMs.

In summary, these works collectively underpin our research direction, emphasizing the role of data alignment and quality over mere volume in the context of fine-tuning LLMs for specialized tasks like autoformalization.

C HOW DOES THIS DESIGN DIRECTLY TEST OUR THESIS?

By following this design, we will obtain quantitative data for both our dependant and independent variables. We will then be able to easily examine the relationship between the alignment coefficient of a data set and the resulting scores on AF benchmarks, allowing us to test whether a higher alignment coefficient will truly result in higher performance of AF. We may even be able to find an expression that predicts the performance of an LLM trained on a data set of certain size given its alignment coefficient with the benchmark it will be tested on.

D PERPLEXITY LOSS FORMULA

Perplexity is calculated as such:

$$PPL(X) = \exp\left\{-\frac{1}{t}\sum_{i}\log p_\theta(x_i|x_{<i})\right\}$$

$\log p_\theta(x_i|x_{<i})$ is the log-likelihood of the ith token conditioned on the preceding tokens $x_{<i}$.

E DATA PREPROCESSING

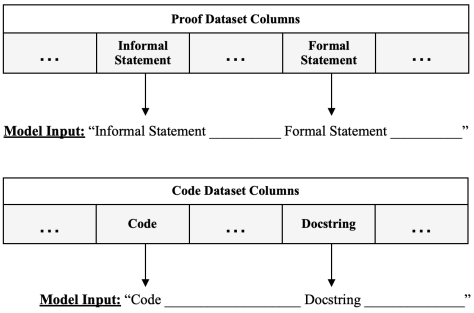

Figure 3: data preprocessing visualization

Figure 3 illustrates how from each column of the datasets we are only interested in two of them for training our model. For proof datasets we are interested in "Informal Statements" and "Formal Statements"; for code datasets we are interested in "code" (the actual function) and "Docstring" (the function header). We ignore the rest of the columns indicated by the ellipses. Then, we feed the model rows of inputs where "Model input" represents a single row of input to the model.

F DATASETS AND THEIR CORRESPONDING NUMBER OF TOKENS

In order to test our claim that an LLM will be better able to perform AF when fine-tuned on a dataset that is closely aligned to the AF benchmark, we must finetune LLMs on datasets of differing alignment to the benchmark. This allows us to observe a relationship between alignment and perplexity loss.

We strategically chose the following datasets to run our experiment on in order to ensure a range of alignment values in our results:

1. AF Dataset (AF): A dataset consisting of informal statements and their formal counterparts in Isabelle designed for training LLMs to perform Autoformalization. We are using it's test set as a benchmark in LLM performance of statement Autoformalization. Thus, we also believe it will result in the lowest perplexity among the proof datasets when used to train an LLM for AF (Miranda, 2021)
2. Destructed AF Dataset (AF-split): This dataset is composed of the AF Dataset's formal and informal statements but the two are split into different lines so that the LLM trains on data that does not explicitly indicate a relationship between the two; we expect this to still obtain a relatively low perplexity score given it's high alignment.
3. The Stack Smol Python Docstrings dataset (Docstring): a dataset consisting of a concise function headers written in informal language and their implementations in python; we use it as a benchmark to assess how well coding datasets can finetune for Autoformalization. (Bird, 2023a)
4. The Stack Dedup Python Docstrings 1.0 percent unified dataset (Docstring 2): A dataset consisting of function headers written in informal language and their implementations in python; given its nature we anticipate it scoring among the lowest of perplexity scores against the Docstring benchmark (Bird, 2023b)
5. C4-EN-10K Dataset (C4): A ten-thousand-entry subset of a database composed of text pulled from Common Crawl (an internet archive) meant for training NLP. Given it's entries are all informal statements not related to mathematics, we predict a high perplexity score in performing AF (Raffel et al., 2019)
6. wikitext-2-raw-v1 Dataset (Wikitext): A subset of the Wikitext dataset; Wikitext is a dataset composed of text taken from Wikipedia pages that met the score guidelines to qualify as either a 'good' or 'featured' article; given it's nature and lack of relevance to AF, we expect a high perplexity score. (Merity et al., 2016)

7. MiniF2F-lean4 Dataset: A subset of the MiniF2F dataset which is comprised of math exercise statements and their formal counterparts in lean; given that it is in a different formal language, we expect a mid-range perplexity score. (Zheng et al., 2021)

8. Proofnet Dataset (Proofnet): This dataset is comprised of statements taken from undergraduate math courses and their formal counterparts in lean; given their similarities, we expect MiniF2F and Proofnet to score similarly in perplexity.(Azerbayev et al., 2023)

9. HumanEvalPack: This dataset consists of a prompt describing a function and implementations of the function in Python, JavaScript, Java, Go, C++, and Rust as well as buggy solutions to serve as bad examples. We expect it to obtain a mid-range score against the Docstring benchmark. (Muennighoff et al., 2023)

Dataset	**Number of Tokens**
AF	4092
C4	4096
Wikitext	4186
Proofnet	4032
MiniF2F	4186
ProofPile	4096
Docstring	4116
Humanevalpack	4004
Docstring2	3790

Table 1: All datasets and their corresponding number of tokens

G PROOF DATASET ALIGNMENT SCORES

Datasets	**Alignment Score**
AF-AF	0.9452813267707825
AFSplit-AF	0.739759624004364
AF-Proofnet	0.6674373149871826
AF-Docstring	0.6128289103507996
AF-MiniF2F	0.5514505505561829
AF-C4	0.3249419331550598
AF-Wikitext	0.26609545946121216

Table 2: Alignment scores of proof datasets on AF benchmark

H PROOF DATASET PERPLEXITY LOSS

Model	**Perplexity**
Standard GPT-2	78.7413
AF Fine Tuned	41.8261
Proofnet Fine Tuned	67.8906
MiniF2F Fine Tuned	71.8377
C4 Fine Tuned	87.4636
Wikitext Fine Tuned	94.9470
Docstring Fine Tuned	75.4504

Table 3: Perplexity Loss for Models Fine-Tuned on Proof Datasets

I CODE DATASET ALIGNMENT SCORES

Datasets	Alignment Score
Docstring-Docstring	0.9609753489494324
Docstring-Docstring2	0.6150134205818176
Docstring-Humanevalpack	0.6194539666175842
Docstring-AF	0.6128289103507996
Docstring-Proofnet	0.5982948541641235
Docstring-MiniF2F	0.5592770576477051
Docstring-C4	0.30464035272598267
Docstring-Wikitext	0.25793445110321045

Table 4: Alignment scores of code datasets on Docstring benchmark

J CODE DATASET PERPLEXITY LOSS

Model	Perplexity
Standard GPT-2	14.7787
Docstring Fine Tuned	11.3641
AF Fine Tuned	15.5574
Proofnet Fine Tuned	14.4500
MiniF2F Fine Tuned	14.0300
C4 Fine Tuned	15.0797
Wikitext Fine Tuned	15.4697
Humanevalpack Fine Tuned	12.7828
Docstring2 Fine Tuned	12.4401

Table 5: Perplexity Loss for Models Fine-Tuned on Code Datasets

K EXPERIMENTS TO FURTHER REINFORCE OUR CLAIMS AND JUSTIFY ASSUMPTIONS

K.1 EXPERIMENT TO VERIFY THAT EACH SUBSET WILL HAVE A SIMILAR PERPLEXITY LOSS TO THAT OF THE ENTIRE DATASET

In our experiments, we evaluated the perplexity loss of various subsets of the dataset on a distinct and separate test set. This approach was crucial to ensure that our findings on perplexity loss accurately reflect the model's ability to generalize and perform on new, unseen data. However, to address the concern that one subset of the training dataset may not reflect the behavior of the entire dataset, we fine-tuned on various subsets of the C4 dataset and studied the perplexity loss each of the subsets resulted when evaluated on the test set.

For each subset (e.g., C4 Subset 1, C4 Subset 2, etc.), we maintained a consistent token size of approximately 4000 tokens. The perplexity scores were then calculated for these subsets using their respective test sets. The results are as follows:

Subset	Number of Tokens
C4 Subset 1	4096
C4 Subset 2	4032
C4 Subset 3	4080
C4 Subset 4	3990

Table 6: subsets and their corresponding number of tokens

Now, we need to calculate the perplexity score for each of these subsets exactly as outlined in the *Evaluation* section. Here are the results:

C4 Subset	Perplexity
Subset 1	87.4636
Subset 2	84.4889
Subset 3	85.9207
Subset 4	87.4829

Table 7: Perplexity Scores for C4 Fine Tuned Model

Here is the graph of all the subsets of C4 along with our original proof dataset fine tuned models:

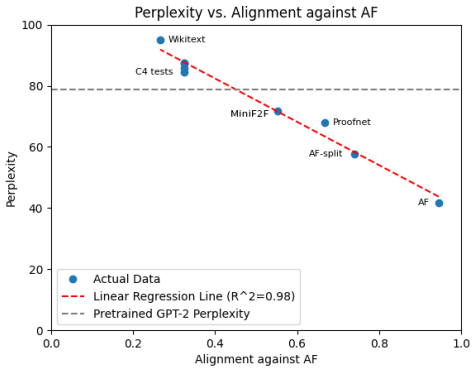

Figure 4: Alignment scores plotted against perplexity loss suggests a linear negative correlation and mirrors our expected findings we described in the Evaluation Design

K.1.1 Discussion of C4 Subset Experiment Results

As seen in *Table 6*, each subset of C4 has comparable perplexity scores. This is further highlighted in the graph where we can see that the subsets are all closely clustered together; this does not affect our line of regression significantly and our claim still holds.

We are making an assumption that the alignment of a subset of the dataset is comparable to the alignment of the entire dataset itself. Hence, our alignment calculations are carried out based on the entire dataset rather than the subset of which we are calculating the perplexity loss. The general intuition is that we want to show a negative relationship between alignment and perplexity which makes the basis of this assumption valid nonetheless. This experiment serves as a proof-of-concept that a subset of a dataset can be used to approximate the subset of the entire dataset.

K.2 Experiment on splitting Formal and Informal Statements in the Training Process:

So far we have pre-processed our data as depicted in *Figure 2*, where each input contains a formal and informal statement (proof dataset) or code and docstring (code dataset). However, we conducted an experiment to observe if inputting formal and informal statements as separate inputs and training on that would produce better results. *Figure 5* below depicts what this would look like:

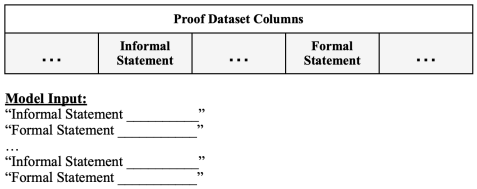

Figure 5: split experiment data preprocessing visualization

We compared the results of AF and AF-Split as follows. We first standardized the number of tokens to 4000 as seen in *Table 8*.

Subset	Number of Tokens
AF Original	4092
AF Split	3960

Table 8: AF and AF Split Tokens

Then, we calculated the alignment as shown in *Table 9*.

Datasets	Alignment Score
AF-AF	0.9452813267707825
AF-AF Split	0.7397596240043640

Table 9: AF and AF Split Alignment Scores

Finally, we Fine-Tuned the model on AF-Split and compared the perplexity loss to AF; this is depicted in *Table 10*.

Model	Perplexity
AF Fine Tuned	41.8261
AF Split Fine Tuned	57.8004

Table 10: Perplexity Loss Scores for AF and AF Split

K.2.1 Discussion of AF-Split Experiment Results

We found that AF-Split had a lower alignment by about 21.7% which is a moderate difference. We also found that the perplexity loss was higher in AF-Split by 38.2% which is also significant. This shows that the model performs better when trained on data that contains packages of related information as opposed to separated inputs of that same related content, at least in the context of autoformalization. For example, the model performs the task of autoformalization better when it knows that an informal statement is somehow related to a formal statement (AF Fine Tuned). This is true because each of the model's inputs is formatted as such: "Informal Statement _____ Formal Statement _____", which implies a general relationship between the two. However, when informal statements and formal statements are separated and passed into the model and the model does not know that they both have some relation to each other (AF-Split Fine Tuned), then it performs the task of autoformalization less accurately.

L Limitations

This research provides valuable insights into the role of data alignment in fine-tuning LLMs for autoformalization tasks. However, several limitations warrant further discussion:

Subset Representation: Our experiments primarily utilized subsets of larger datasets due to computational constraints. While these subsets were chosen to be representative, they may not capture

the full diversity and complexity of the complete datasets. This limitation raises concerns about the generalizability of our findings across entire datasets.

Internal Dataset Diversity: The study focused on data alignment but did not extensively explore the impact of internal dataset diversity on model performance. Diversity within datasets, in terms of varied linguistic structures and concepts, could significantly influence model training and effectiveness, an aspect that remains unexamined in our current work.

Computational Resources: The scale of our experiments was constrained by the available computational resources. This limitation impacted our ability to train models on the full version of datasets and to experiment with larger, more complex LLM architectures. As a result, the scalability and applicability of our findings to more extensive training scenarios might be limited.

Unexplored Factors: Additional factors, such as the influence of different preprocessing techniques, hyperparameter tuning, and the choice of baseline models, were not extensively explored. These elements could have a notable impact on the outcomes of LLM training and merit further investigation.

Future Directions: Addressing these limitations presents opportunities for future research. Studies involving more comprehensive datasets, diversified in terms of size, structure, and content, would be valuable. Additionally, expanding computational capabilities to accommodate larger models and extensive training regimes would further enhance the understanding of LLM performance in relation to data alignment.