Beyond Accuracy Optimization: Computer Vision Losses for Large Language Model Fine-Tuning

Anonymous ACL submission

Abstract

Large Language Models (LLMs) have demonstrated impressive performance across various tasks. However, current training approaches combine standard cross-entropy loss with extensive data, human feedback, or ad hoc methods to enhance performance. These solutions are often not scalable or feasible due to their associated costs, complexity, or resource requirements. This study investigates the use of established semantic segmentation loss functions in natural language generation to create a versatile, practical, and scalable solution for fine-tuning different architectures. We evaluate their effectiveness in solving Math Word Problems and question answering across different models of varying sizes. For the analyzed tasks, we found that the traditional Cross-Entropy loss represents a sub-optimal choice, while models trained to minimize alternative (task-dependent) losses, such as Focal or Lovász, achieve a mean improvement of +42% on exact match without requiring additional data or human feedback. These findings suggest a promising pathway for more efficient and accessible training processes.

1 Introduction

Generative Language Models have shown impressive capabilities across various scenarios (Raffel et al., 2020). Recent advancements in Large Language Models have made this even more evident (Liang et al., 2022). However, the performance of these models is influenced by three main factors: model size, amount of training data, and training strategy (Luo et al., 2023; Yue et al., 2024). Increasing model size requires more computational resources while training on vast data collections is essential to achieve competitive results when increasing the size. Additional training refinements have been introduced recently, some of which involve human experts providing feedback to enhance model performance, as in Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2023) where human preferences are then used to align the model outputs.

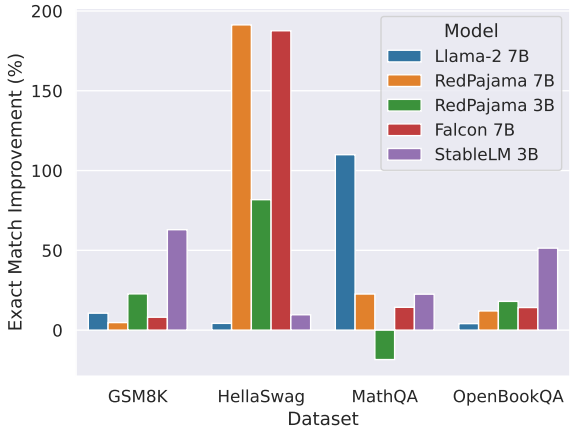

Figure 1: Percentage of improvement using the best loss (among Focal, Lovász, Generalized Dice, and Self-Adjusting Dice) for the task with Cross-Entropy compared to model fine-tuned with Cross-Entropy only.

Despite the improved performance, developing these models requires massive amounts of resources, power, time, and therefore significant costs, making them accessible only to very few leading companies. The increasing costs and resource requirements have already led to the development of solutions aimed at democratizing the training of these models. An example of this is the use of Parameter Efficient Fine-Tuning (PEFT) (Mangrulkar et al., 2022) like Low-Rank Adaptation (LoRA) (Hu et al., 2021a) or derived strategies, often combined with quantization techniques, to enable lightweight fine-tuning of these computationally expensive architectures. Some works (e.g., Zephyr (Tunstall et al., 2023)) circumvent the need for human feedback by distilling the knowledge of larger and more powerful models (e.g., GPT-4), leveraging the so-called AI Feedback (AIF). However, this may cause the propagation of potential

biases from the larger model, or the generation of unfactual content, potentially resulting in a less accurate representation of reality (Liu, 2023; Hosking et al., 2024). Although Direct Preference Optimization (DPO) (Rafailov et al., 2023) and similar methods address the instability in reinforcement learning training (Rafailov et al., 2023; Liu et al., 2022) and the computational requirements, their effectiveness is limited by the quality of the prior supervised fine-tuning stage (Xu et al., 2024b; Tunstall et al., 2023).

Supervised fine-tuning still remains the most memory-efficient alternative, however the collection of fine-tuning and instruction-tuning datasets presents similar challenges, as it requires additional annotation costs that may not always be accessible, especially when dealing with multiple datasets to annotate (Yue et al., 2024). While costs can be mitigated by relying on weak annotation, similar to possible issues of using AIF, the quality of annotations is not always guaranteed. The need for larger models and additional data pertains not only to general-purpose systems (e.g., ChatGPT) but also to task-specific models for more complex scenarios, as in the case of Math Word Problems (MWP). Specifically, state-of-the-art solutions often employ more complex training procedures, involving multiple training stages (e.g., supervised fine-tuning, instruction tuning, preference-based tuning) (Luo et al., 2023; Yue et al., 2024; Azerbayev et al., 2024) to achieve improved performance. However, similar to general-purpose models, this leads to more expensive solutions. Additionally, these approaches are also less portable since they are tailored to one specific task.

To tackle the above-mentioned challenges, in this work we raise some concerns about the standard practice of cross-entropy (CE) loss optimization, which is the usual language modeling objective, and we show that a more accurate selection of the loss function to optimize can be incredibly beneficial for model training. Specifically, by using a loss function tailored to the task under analysis and leveraging LoRA, we effectively fine-tune LLMs with small amounts of data. The underlying idea is that for certain language tasks, it is desirable to optimize not only for the correctness of the output but also for the structural adherence of the generated text to a specific format or syntax. This is particularly relevant for tasks involving formal languages or well-defined procedures, such as mathematical reasoning, where the intermediate steps and reasoning process must follow a strict structure. Consequently, we hypothesize that accounting for these characteristics by employing a more appropriate loss function could lead to improved performance.

Our solution does not involve either the implementation of complex training procedures (e.g., (Luo et al., 2023)), further pre-training (e.g., (Azerbayev et al., 2024)), distilling knowledge from larger models (e.g., (Tunstall et al., 2023)) or collecting human preferences (OpenAI, 2023). In contrast, our approach focuses on selecting a more suitable loss function based on the task at hand, achieving improvements over the standard cross-entropy loss through a single training stage, as shown in Figure 1. In this work, we focus on mathematical reasoning and closed-question answering, which are common benchmark tasks (Liang et al., 2022; et al, 2023; Hendrycks et al., 2021) that have a clear and well-defined expected output. Furthermore, we choose these tasks since we claim that, for both of them, the role of human preferences is secondary since it is difficult to "prefer" one output with respect to another, especially in mathematical reasoning where there could be more than one procedure to get the final solution.

Our results show that accurately choosing the right loss function (combined with Cross-Entropy) can improve performance on the analyzed tasks using the same amount of data without adding complexity to the training process.

The source code to reproduce the experiments is available for research purposes at `https://anonymous.4open.science/r/segmentation-losses-nlp-5B73`.

2 Related Works

The evolution of Large Language Models has been driven by various innovative training methods. This section provides an overview of the existing approaches for training LLMs, highlighting the challenges and benefits of each approach. Additionally, we explore the development of alternative loss functions beyond cross-entropy in both natural language processing and computer vision fields.

2.1 Training methods for Large Language Models

The most common approaches to training LLMs are pre-training, instruction tuning, supervised fine-tuning, and tuning by preferences.

Pre-Training. Among these, effective pre-training remains a key solution for achieving the best results (Azerbayev et al., 2024; Jiang et al., 2023; Zhou et al., 2023). However, the need for high computational resources and a large amount of usable data (e.g., the source license must grant permission for the intended scope) makes this solution not always scalable or feasible in most cases.

Supervised Fine-tuning and Instruction Tuning. Supervised fine-tuning and instruction tuning are common solutions to adapt pre-trained models to a series of tasks (Xu et al., 2024a; Yue et al., 2024; Jiang et al., 2023), as this approach requires less data and can exploit efficient solutions like LoRA (Hu et al., 2021a) and quantization to scale the training. However, there is still a need for large data collections since the employed language modeling loss (commonly Cross-Entropy) does not effectively represent the salient parts of the instructions (e.g., it may not correctly represent the token distribution (Lin et al., 2017)). In many cases, ad hoc collections must be created, and since costs and time are still high, many solutions leverage other language models to create annotations (Yue et al., 2024; Lian et al., 2023; Yu et al., 2023). Although this is a more cost-effective solution than human annotation, it could lead to biased datasets (Tan et al., 2024).

Human Feedback. RLHF (Christiano et al., 2023) and DPO (Rafailov et al., 2023) propose new methods to train models using human preferences.

Human feedback has proven helpful in tasks that require evaluating the model's text generation capability (Stiennon et al., 2020; Fan et al., 2019; Ethayarajh et al., 2022), where quantitative evaluation alone cannot cover all aspects of the desired output (Chang et al., 2024). However, Zhou et al. (2023) highlights the limitations of RLHF, while Hosking et al. (2024) argues that preference scores under-represent crucial aspects such as factuality, which is an objective for question-answering and mathematical reasoning.

Moreover, this approach requires collecting human preferences, which is costly. In this case, some solutions use distilled feedback to avoid extra costs (Tunstall et al., 2023), although this exposes them to potential biases of the employed model.

WizardMath (Luo et al., 2023) proposes a different approach to include feedback in mathematical problems named Reinforcement Learning from Evol-Instruct Feedback (RLEIF). Although human feedback (with its potential biases) is avoided, RLEIF faces scalability issues due to the need for training two additional models (i.e., Instruction Reward and Process-Supervised Reward models) to produce various feedback types.

2.2 Loss functions beyond Cross-Entropy

Despite the most common approaches involving cross-entropy and the optimization of feedback through RL, other methods exist. Reinforcement learning has already been used to optimize the BLEU metric (Ranzato et al., 2015; Wang et al., 2019), before the employment of feedback. However, the training instability and the unclear contribution in some settings (Wu et al., 2018) are great drawbacks, coupled with its non-differentiability. EISL loss (Liu et al., 2022) was proposed to optimize the n-grams matching in a differentiable and more stable way, but it is applicable to non-autoregressive models. Self-Adjusting Dice Loss (Li et al., 2020), a combination of Dice and Focal losses, was proposed to address imbalanced classification tasks in NLP. However, they employed encoder-only architectures, and the benefits depend on the specific task. Dice and Focal losses originate from the computer vision field (in particular semantic segmentation), which is rich in loss functions designed to address class imbalance (which translates to token imbalance in NLG) and effectively penalize prediction errors. Dice (Milletari et al., 2016), Generalized Dice (Sudre et al., 2017), Focal (Lin et al., 2017), and Lovász (Berman et al., 2018) are some established loss functions that aim to address these issues by optimizing objectives other than accuracy (e.g., Dice score, Intersection-over-Union), unlike Cross-Entropy (Li et al., 2020). Additionally, their combination has proven to be more effective than employing them singularly in computer vision (Taghanaki et al., 2019; Yeung et al., 2022; Iantsen et al., 2021; Hu et al., 2021b).

Transferring this approach to causal language modeling is particularly appealing since these loss functions are differentiable, stable during training, and generalizable to many tasks. Although existing solutions for causal language modeling have tried to improve the training in different ways, they still suffer from scalability problems in terms of data, training time, and costs. This work aims to propose a simple, generalizable, and scalable approach to improve existing models without involving large data collection or complex training strategies. We show that a better extraction of knowledge from

existing data can already provide relevant results' improvements by training only a few parameters (using LoRA) and small data collections (between 500 and 40K samples).

3 Methodology

In this section, we formally introduce the loss functions we employ, shortlisted from the classification presented in Ma et al. (2021), and explain their rationale. We describe our approach when employing them for causal generation. For the sake of readability, all loss formulations are reported in Appendix A.

3.1 Distribution-based losses

This family of loss functions is derived from the Kullback-Leibler Divergence. They aim to optimize the model weights according to the differences between the observed and expected distributions.

Cross Entropy Loss. Cross-Entropy (CE) is an accuracy-oriented function, i.e., it aims to maximize the accuracy (AC) metric globally in the predicted tokens (Li et al., 2020). CE is the most established loss for pre-training and fine-tuning language models. Cross-entropy does not consider the underlying structures of predictions or any differences between classes and errors. Class imbalance is common in language problems, where classes are represented by tokens in the vocabulary, and token distributions are rather diversified (see Appendix C). Although weighted cross-entropy may address this issue, assigning a proper weight to each class (i.e., token) can be challenging.

Focal Loss. Focal Loss (FL) (Lin et al., 2017) is a variant of CE that is specifically designed to address the class imbalance problem. It aims to reduce the relative loss for well-classified examples while emphasizing training on hard, misclassified ones. Although Focal loss does not directly consider the class distribution, it automatically distinguishes between hard and easy samples. This proves beneficial in correctly predicting underrepresented classes. Notably, this solution gives more importance to errors (i.e., wrongly predicted tokens) than cross-entropy.

3.2 Region-based losses

This family of loss functions optimizes the model weights according to the differences between two mathematical sets.

Dice Loss. It is the main representative of the region-based loss family. Dice Loss (DL) (Milletari et al., 2016) optimizes the Dice Score (DS) between two sets. DL directly maximizes a soft version of the Dice Score. It assigns different weights to errors and correct predictions. However, correct predictions are deemed more relevant than wrong predictions; therefore, errors may not be sufficiently penalized.

Generalized Dice Loss. A generalization of the Dice score (Crum et al., 2006) and the corresponding Generalized Dice Loss (GDL) was proposed to consider each class's volume. This formulation proposes to self-adjust the weight of each class for each sample to address the class imbalance issue.

Lovász Loss. Lovász Loss (LL) (Berman et al., 2018) is a surrogate loss deriving from the Jaccard Index (or Intersection-over-Union). LL takes into account both errors and correct predictions. In contrast to Dice loss, which assigns more weight to correctly classified samples, the formulation of Lovász loss allows for an adequate penalty for misclassifications. In many language tasks, the aim is not only to penalize errors but also to force the system to avoid introducing extra tokens or omitting certain tokens. This objective can be reached by optimizing the Jaccard Index. We claim that optimizing this objective can be particularly beneficial for the mathematical reasoning task if we ask the model to generate both the final answer and the intermediate reasoning steps. In this case, the intermediate steps must adhere to a stringent structure in terms of syntax (i.e., Math is a formal language) and content (i.e., there are usually not many alternative procedures to get the final answer). This makes the task suitable for optimization using Lovász loss.

3.3 Compound Losses

Compound losses are created by combining other loss functions, resulting in a more complex (and possibly more representative) objective function.

Self-Adjusting Dice Loss. We also evaluate Self-Adjusting Dice Loss (SADL) (Li et al., 2020), which combines the intuitions of Dice and Focal losses. The rationale behind introducing the Focal component in the Dice Loss is to address the imbalance problem between well-classified and misclassified tokens, which is not adequately covered by Dice loss.

3.4 Loss application to language generation

Let I and A be the instruction and its corresponding answer. Let i and a be the number of tokens in I and A, respectively. We define the language modeling loss as a convex combination (Taghanaki et al., 2019) of CE and one of the various loss functions L under consideration (i.e., FL, DL, GDL, SADL, and LL): $\mathcal{L} = \lambda \text{CE}_{I,A} + (1-\lambda)\text{L}_A$, where $\lambda \in [0, 1]$. CE is applied to both the I's and A's tokens, while L is applied only to the A's tokens of the answer (i.e., ground truth), as shown in Figure 2. This approach emphasizes the actual target sequence of interest, which follows a more rigid structure. Applying the second component to the instruction tokens may wrongly emphasize underrepresented tokens that are not useful in this case.

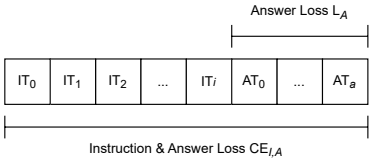

Figure 2: A graphical sketch of how the combined loss is applied to instruction I and answer A. *IT*s are instruction tokens, *AT*s are answer tokens.

3.5 Evaluation Metrics

We employ both standard metrics that consider the final result only (i.e., Exact Match (EM)) and metrics that are specifically tailored to assess the reasoning steps. Since reasoning step metrics are suited to MWP only, they will be reported only for GSM8K and MathQA datasets.

Metrics for the reasoning steps. We adopt the ROSCOE metrics (Golovneva et al., 2022) and general purpose metrics to evaluate the correctness of intermediate reasoning steps, given the systematic and precise nature of mathematical language: Jaccard Index (or IoU, in short) (see Appendix A), and Commutative IoU (C-IoU), which we define as a variant of IoU that accounts for the commutative property of mathematical operations. These metrics are calculated between predicted rationales and ground truth reasoning steps. Unlike ROSCOE, adopting this approach eliminates reliance on external models, thus circumventing potential limitations inherent to the models used.

ROSCOE metrics consider four perspectives: Semantic Alignment (SA), Semantic Similarity (SS), Logical Inference (LI), and Language Coherence (LC). Each metric ranges between zero (worst) and one (best). While, for completeness, we evaluate all the proposed metrics, we argue that LC metrics may not be suitable for assessing mathematical steps, as they are not expressed in natural language.

4 Experimental Results

We perform an extensive experimental evaluation on two tasks for a total of four datasets, five models, and five loss functions. In the following, we summarize the main results reporting the average across models. Detailed results are available in Appendix E.

4.1 Datasets

We selected four datasets, each including at least training and validation sets, neglecting those containing only the test set (being designed for zero-shot benchmarking).

We selected two MWP datasets: GSM8K (Cobbe et al., 2021) and MathQA (Amini et al., 2019). We have chosen these datasets since they include both the final result and the operational annotations (i.e., reasoning steps) leading to the final answer. We also selected two multiple-choice datasets included in the HELM benchmark: OpenBookQA (Mihaylov et al., 2018) and HellaSwag (Zellers et al., 2019). We consider these QA datasets since answers are mainly based on reading comprehension rather than relying on prior knowledge of the LLM.

Detailed information on the considered datasets, including their training/validation/test set splits, are available in Appendix B.

4.2 Competitors

In our selection of competitors, we considered two key criteria: models with comparable sizes in terms of billions of parameters and open-source solutions rather than closed-source alternatives. As shown in Table 1, we selected open-source solutions trained with different strategies: MAmmoTH (Yue et al., 2024), WizardMath (Luo et al., 2023), WizardLM (Xu et al., 2024a), Llemma (Azerbayev et al., 2024), MetaMath (Yu et al., 2023), and Mistral (Jiang et al., 2023). For the sake of completeness, we also included a closed-source state-of-the-art model (GPT-4 (OpenAI, 2023)). Some models employ domain-specific training, while other approaches are more generalist.

These models employ diverse training strategies, including continual pre-training, instruct-following

Model	Training Strategy	# Training Samples	# Params	# Updated Params	Generalist	Domain
Ours	IT	≈ 5K-40K	3B-7B	< 1%	✓	–
MAmmoTH (Yue et al., 2024)	IT	≈ 260K	7B	100%	✗	Math
WizardMath (Luo et al., 2023)	RLEIF	≈ 96K	7B	100%	✗	Math
WizardLM (Xu et al., 2024a)	EI	≈ 286K	7B	100%	✓	–
Llemma (Azerbayev et al., 2024)	P	≈ 30M	7B	100%	✗	Math
MetaMath (Yu et al., 2023)	IT	≈ 395K	7B	100%	✗	Math
Mistral v0.2 (Jiang et al., 2023)	IT	?	7B	100%	✓	–
GPT4 (OpenAI, 2023)	RLHF	?	?	?	✓	–

Table 1: Competitors details. *IT* stands for Instruction Tuning, *P* for pre-training, *RLHF* for Reinforcement Learning from Human Feedback, *EI* for Evol-Instruct, and *RLEIF* for Reinforcement Learning from Evol-Instruct Feedback.

fine-tuning, Reinforcement Learning From Evol-Instruct Feedback, Evol-Instruct, and Reinforcement Learning from Human Feedback. Consequently, these approaches often necessitate larger datasets or human interventions or rely on other language models for training.

The training data sizes vary significantly, ranging from a hundred thousand samples to millions of samples for models like Llemma. While most competitors updated 100% of their parameters during training, our approach involves updating less than 1% of the 3-7 billion parameters using LoRA.

4.3 Models

We employ the following LLMs with a number of parameters ranging from 3B to 7B: RedPajama-Incite-3B (Together Computer, 2023), StableLM-3B (Tow et al., 2023), RedPajama-Incite-7B (Together Computer, 2023), Falcon-7B (Almazrouei et al., 2023), and Llama-2-7B (Touvron et al., 2023). Except for Llama-2 (which is selected as one of the most well-known open-source models), the other ones are selected with the following criteria: (1) They are open-source; (2) They show promising results according to HELM benchmark (Liang et al., 2022); (3) The majority of their training datasets are public or clearly stated to avoid overlapping with the analyzed datasets; (4) We consider only the pre-trained version (without any instruction tuning or tuning by human preferences).

More details about the selected models can be found in Appendix D.

4.4 Experimental settings

We set the number of training steps to around 25000 and the batch size to 2. We employ Low-Rank Adaptation (Hu et al., 2021a), AdamW optimizer (Loshchilov and Hutter, 2017), and a linear learning rate scheduler with a warmup of 500 steps. Further information about the experimental settings and implementation details are given in Appendix G.

Loss	HellaSwag ↑	OpenBookQA ↑	GSM8K ↑	MathQA ↑	MR ↓
CE	47.36	75.60	15.83	5.12	3.33
FL	**71.68**	80.88	15.41	**5.52**	2.43
GDL	47.39	75.08	15.00	5.04	3.93
LL	58.08	**82.80**	**17.76**	4.76	**1.90**
SADL	41.83	67.40	15.91	4.48	3.42

(a) Exact Match

		General purpose		ROSCOE			
	Loss	IoU ↑	C-IoU ↑	SA ↑	SS ↑	LI ↑	LC ↑
GSM8K	CE	15.52	19.27	81.14	65.75	34.91	37.58
	FL	15.09	18.71	81.38	**66.67**	**36.74**	37.60
	GDL	15.15	18.70	81.08	65.73	34.70	37.60
	LL	**17.39**	**21.10**	**81.39**	66.33	36.00	37.46
	SADL	15.64	19.51	81.33	66.29	35.47	**37.62**
MathQA	CE	36.72	36.78	85.12	68.43	24.21	38.86
	FL	33.73	33.79	85.29	68.39	23.75	38.80
	GDL	36.30	36.36	85.07	67.05	21.01	38.90
	LL	**43.25**	**43.31**	**85.76**	**70.03**	**28.68**	38.75
	SADL	34.18	34.23	84.97	67.05	20.42	**38.95**

(b) Reasoning Step metrics

Table 2: Macro-average achieved on analyzed datasets.

4.5 Answer generation results

Considering the exact match, as shown in Table 2a, the CE-only setting is a suboptimal choice in every case. Based on the mean rank across all models and datasets (i.e., the average rank of each loss), the best losses for these tasks are the Focal and Lovász losses. They show a difference of 0.9 and 1.43, respectively, compared to the CE rank.

The effectiveness of the Focal and Lovász losses is likely due to their distinct approaches to handling prediction errors. The Focal loss underestimates the loss contributions of well-predicted samples based on class distribution, while the Lovász loss penalizes wrong predictions without suppressing well-predicted samples according to their distributional behavior.

4.6 Reasoning steps generation results

On both MWP datasets, considering reasoning metrics, the combination with Lovász loss consistently outperforms the CE-only setting as shown in Table 2b. Also in this case, it achieves the best performance, likely thanks to the effect of misclassified sample penalties. Specifically, while cross-entropy and Focal loss aim to maximize global accuracy, LL aims to maximize the global IoU, i.e., it considers both the absence of extra tokens and the presence of missing tokens.

The results on MathQA and GSM8K show that the final answer tends to be wrong in many cases (low EM values), while the reasoning steps tend to be quite accurate (high or medium-high reasoning step metrics). This highlights that the models generally struggle to correctly predict the final result despite showing a good capability in formulating the mathematical reasoning steps.

The complete set of results on all datasets for all models and metrics are available in Appendix E, along with statistical tests for significance between cross-entropy and the other loss functions.

Correlation analysis between reasoning step metrics. We investigate the Pearson's correlation between the ROSCOE metrics, EM, and IoU to understand if the optimization of this last metric goes in the same direction as more complex ones. As expected, IoU is positively correlated (values in range $[0.5, 0.7]$) with many Semantic Alignment metrics, as Reasoning Alignment, External Hallucination, Redundancy, Common Sense Error, Missing Step, and with a Sematic Similarity metric, i.e., Semantic Coverage Chain. This confirms that optimizing IoU (through the Lovász loss) is a reasonable proxy to optimize the reasoning step metrics. More details are given in Appendix F.

Error type analysis in MWP. We analyze the most common mistakes observed in MWP reasoning steps. We consider the following metrics covering complementary types of reasoning errors[1]:

- Extra Step (ES): proportion of predicted rationales not included in the gold annotations: ES $= |PS - GTS|/|PS|$
- Missing Step (MS): proportion of gold rationales not generated by the model: MS $= |GTS - PS|/|GTS|$
- Wrong Operators (WO): proportion of predicted rationales with correct operands but wrong sign according to the gold rationales: WO $= |PS_{wo}|/|E|$
- Inverted Operands (IO): proportion of predicted rationales in which the operands have an incorrect position, considering non-commutative operations: IO $= |PS_{io}|/|E|$

where GTS and PS are the ground truth and predicted reasoning steps, PS_{wo} and PS_{io} are predicted steps with a wrong operator and inverted operands, and E is the set of errors, i.e., the set of predicted reasoning steps that do not match the gold rationales.

The results are summarized in Table 3. Lovász loss yields the lowest percentages of errors across most error types, particularly in reducing the amount of missing steps. The errors related to wrong operators and inverted operands affect approximately only 4-5% of the reasoning steps for all loss functions. Overall, generating fully accurate reasoning chains remains challenging, but losses such as Lovász loss can help mitigate certain types of errors, making it a preferable training loss to cross-entropy.

Loss	ES ↓	MS ↓	WO ↓	IO ↓
CE	67.60%	67.78%	4.68%	5.13%
FL	67.85%	68.48%	**4.22%**	**4.66%**
GDL	68.30%	66.95%	4.57%	5.00%
LL	**62.87%**	**62.83%**	4.27%	**4.66%**
SADL	70.40%	67.32%	4.71%	5.21%

Table 3: Mean errors in mathematical reasoning over all models and datasets.

4.7 Results on a reduced number of samples

We evaluate the effectiveness of the proposed approach on each task and dataset using our best model (i.e., StableLM) by reducing the number of training samples to 40% and 10%, while also reducing the training duration by the same amount. In Table 4, we present the average results on each dataset by loss. We show that cross-entropy does not generally yield satisfactory results when the amount of data is reduced. Conversely, losses such as Focal and Lovász demonstrate better capability in extracting desired knowledge even from fewer samples. The trend is the same for both exact match and reasoning step metrics.

[1]To the best of our knowledge, there are no standard metrics to evaluate mathematical reasoning.

	Dataset	CE	GDL	FL	LL	SADL
10%	HellaSwag	71.90	70.71	**79.80**	79.17	77.77
	OpenBookQA	**80.40**	75.00	79.80	75.80	74.00
	GSM8K	13.72	13.57	13.57	**15.31**	13.27
	MathQA	5.61	5.54	**6.59**	6.03	5.95
40%	HellaSwag	81.72	82.05	**90.82**	86.78	84.07
	OpenBookQA	83.00	83.00	**84.60**	83.00	82.20
	GSM8K	21.68	21.76	23.88	**26.08**	20.77
	MathQA	7.08	5.12	**8.29**	7.80	3.47

(a) Exact Match

	Dataset	CE	GDL	FL	LL	SADL
10%	GSM8K	13.63	13.55	13.33	**14.59**	13.24
	MathQA	9.37	8.85	10.54	**10.66**	8.85
40%	GSM8K	18.98	18.72	19.73	**21.86**	18.11
	MathQA	35.70	36.88	37.04	**40.01**	34.61

(b) Intersection-over-Union

Table 4: Results of the best-performing model on different training dataset subsets (10% and 40%).

4.8 Comparison between CE-Only and Loss-By-Task Instruction Tuning

To evaluate the effectiveness of our approach in an instruction-tuning scenario (similar to (Yue et al., 2024)), we train our model on a combined dataset containing task-specific samples from all previously mentioned datasets. We compare the results achieved by cross-entropy alone and combined with the other loss functions considered. According to previous sections, we selected Lovász for MWP and Focal for QA. The results in Table 5 confirm our strategy is still effective in a dataset composed of different tasks.

Loss	HellaSwag	OpenBookQA	GSM8K	MathQA
CE	37.69	41.08	10.06	3.28
Loss-By-Task	**66.92**	**49.31**	**11.77**	**3.84**

(a) Exact Match

	GSM8K				MathQA			
Loss	SS ↑	SA ↑	LI ↑	LC ↑	SS ↑	SA ↑	LI ↑	LC ↑
CE	66.95	81.13	38.25	**37.80**	72.02	84.76	30.26	39.25
Loss-By-Task	**67.02**	**81.23**	**38.94**	37.66	**72.84**	**84.80**	**32.09**	**40.16**

(b) ROSCOE metrics

Table 5: Mean performance over all datasets in Instruction Tuning mode.

4.9 Comparison with the state of the art

As shown in Table 6, our proposed model achieves the best results in 2 out of 4 datasets. In contrast, domain-specific models, such as MAmmoTH and Llemma, experience a notable degradation in performance when evaluated on closed-ended QA datasets. Our proposed approach achieves comparable performance to WizardMath according to the mean rank (MR), proving its effectiveness across various scenarios without employing any additional steps after fine-tuning (e.g., tuning by preferences). Regarding rationale generation, our best model ranks in the top 3 positions according to the mean rank. Although WizardMath and Mistral are the best-performing in terms of exact match, they exhibit the lowest performance according to the ROSCOE metrics. This confirms the fact that providing the right answer does not necessarily imply the correct reasoning.

Model	GSM8K ↑	MathQA ↑	HellaSwag ↑	OpenBookQA ↑	MR ↓
Our Best	28.66	10.06	**85.69**	**87.20**	2.33
MAmmoTH	37.76	15.51	7.30	3.60	4.00
WizardMath	46.10	**32.43**	36.84	60.00	**2.25**
WizardLM	9.02	3.84	26.81	33.40	5.50
Mistral	19.64	9.76	49.26	74.40	3.50
Llemma ⑧	30.33	9.53	24.47	21.20	5.00
MetaMath	**60.27**	14.43	14.56	19.20	4.25
GPT4 ✻	93.20	–	95.30	96.00	–

(a) Exact Match

	GSM8K				MathQA				
Model	SS ↑	SA ↑	LI↑	LC ↑	SS ↑	SA ↑	LI ↑	LC ↑	MR ↓
Our Best	66.10	81.76	35.70	24.10	**67.03**	**86.14**	24.30	25.93	3.38
MAmmoTH	66.46	81.02	10.29	24.48	64.70	80.02	17.45	23.45	4.88
WizardMath	64.18	80.18	15.18	27.36	63.36	79.71	5.60	27.88	4.00
WizardLM	63.71	80.05	11.44	**27.45**	64.25	79.94	14.63	27.48	4.75
Mistral	63.56	81.13	13.87	26.26	62.97	80.52	10.58	26.76	4.75
Llemma ⑧	**74.50**	**85.70**	**46.74**	25.36	61.96	79.32	**66.59**	**36.22**	**3.00**
MetaMath	66.71	82.50	35.53	26.04	64.80	80.01	20.50	26.26	3.25
GPT4 ✻	–	–	–	–	–	–	–	–	–

(b) ROSCOE metrics

Table 6: Competitors results on analyzed datasets. ✻ indicates results taken from other papers (Liang et al., 2022; OpenAI, 2023) and ⑧ indicates model tested in 8-shots. The **best**, second-best, and third-best results are indicated in each column.

5 Conclusion and Future Work

In our work, we applied semantic segmentation losses to improve the fine-tuning of LLMs for mathematical reasoning and closed-ended question-answering. Our results show that using appropriate loss functions during fine-tuning can boost performance without extra data or human feedback. In practice, this suggest a promising pathway for more efficient and accessible training processes. Future work will focus on designing new task-specific loss functions and exploring other tasks.

Limitations

We analyzed only English datasets from the mathematical reasoning and reading comprehension domains. Additional experiments on other languages and tasks would strengthen the generalizability of

our findings. It is worth noting that we limited our analysis to existing loss functions in computer vision, which may be suboptimal for the tasks under consideration. We focused on tasks with strong constraints to verify the effectiveness of the analyzed loss functions; however, this approach may pose limitations in datasets with more open-ended solutions lacking well-defined patterns.

Our model choice was based on the available resources, and we tested only 3B and 7B models. Although we could expect similar findings with larger models, we cannot confirm this claim.

Ethics Statement

From our understanding, the datasets employed in this study do not contain any personal information, but they can contain some harmful or inappropriate content. This claim can be extended to the employed models, which could provide non-factual, biased, harmful, or inappropriate answers. Their usage is subject to the limitations stated in their respective technical reports and licenses. The generated answers are not intended to offend or harm anyone. Language models have environmental impacts due to the high computing requirements during pre-training and fine-tuning. We have made efforts to be computationally responsible by reusing open-sourced pre-trained models and employing efficient fine-tuning methods such as LoRA (Hu et al., 2021a). The gains from improved losses help amortize the resource costs over higher utility. Overall, we have made reasonable efforts to ensure the transparency and auditability of our experimental methodology.

References

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. The falcon series of open language models.

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. MathQA: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen Marcus McAleer, Albert Q. Jiang, Jia Deng, Stella Biderman, and Sean Welleck. 2024. Llemma: An open language model for mathematics. In *The Twelfth International Conference on Learning Representations*.

Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. 2018. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4413–4421.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2024. A survey on evaluation of large language models. *ACM Trans. Intell. Syst. Technol.*, 15(3).

Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2023. Deep reinforcement learning from human preferences.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

W.R. Crum, O. Camara, and D.L.G. Hill. 2006. Generalized overlap measures for evaluation and validation in medical image analysis. *IEEE Transactions on Medical Imaging*, 25(11):1451–1461.

Thomas G. Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923.

Aarohi Srivastava et al. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. 2022. Understanding dataset difficulty with $\mathcal{V}$-usable information. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5988–6008. PMLR.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: long form question answering. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3558–3567. Association for Computational Linguistics.

Olga Golovneva, Moya Peng Chen, Spencer Poff, Martin Corredor, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. 2022. Roscoe: A suite of metrics for scoring step-by-step reasoning.

In *The Eleventh International Conference on Learning Representations*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Tom Hosking, Phil Blunsom, and Max Bartolo. 2024. Human feedback is not gold standard. In *The Twelfth International Conference on Learning Representations*.

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

Xiaoling Hu, Yusu Wang, Li Fuxin, Dimitris Samaras, and Chao Chen. 2021b. Topology-aware segmentation using discrete morse theory. In *International Conference on Learning Representations*.

Andrei Iantsen, Dimitris Visvikis, and Mathieu Hatt. 2021. *Squeeze-and-Excitation Normalization for Automated Delineation of Head and Neck Primary Tumors in Combined PET and CT Images*, page 37–43. Springer International Publishing.

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. 2023. Mistral 7b.

Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. 2020. Dice loss for data-imbalanced NLP tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 465–476, Online. Association for Computational Linguistics.

Wing Lian, Bleys Goodson, Eugene Pentland, Austin Cook, Chanvichet Vong, and "Teknium". 2023. Openorca: An open dataset of gpt augmented flan reasoning traces. `https://https://huggingface.co/Open-Orca/OpenOrca`.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. *ACL*.

Gabrielle Kaili-May Liu. 2023. Perspectives on the social impacts of reinforcement learning with human feedback.

Guangyi Liu, Zichao Yang, Tianhua Tao, Xiaodan Liang, Junwei Bao, Zhen Li, Xiaodong He, Shuguang Cui, and Zhiting Hu. 2022. Don't take it literally: An edit-invariant sequence loss for text generation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2055–2078, Seattle, United States. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *arXiv preprint arXiv:2308.09583*.

Jun Ma, Jianan Chen, Matthew Ng, Rui Huang, Yu Li, Chen Li, Xiaoping Yang, and Anne L Martel. 2021. Loss odyssey in medical image segmentation. *Medical Image Analysis*, 71:102035.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.

Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. 2016. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571.

OpenAI. 2023. GPT-4 technical report. *ArXiv*, abs/2303.08774.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence level training with recurrent neural networks. *CoRR*, abs/1511.06732.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021.

Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. 2017. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*, pages 240–248. Springer.

Saeid Asgari Taghanaki, Yefeng Zheng, S Kevin Zhou, Bogdan Georgescu, Puneet Sharma, Daguang Xu, Dorin Comaniciu, and Ghassan Hamarneh. 2019. Combo loss: Handling input and output imbalance in multi-organ segmentation. *Computerized Medical Imaging and Graphics*, 75:24–33.

Zhen Tan, Alimohammad Beigi, Song Wang, Ruocheng Guo, Amrita Bhattacharjee, Bohan Jiang, Mansooreh Karami, Jundong Li, Lu Cheng, and Huan Liu. 2024. Large language models for data annotation: A survey.

Together Computer. 2023. Redpajama: An open source recipe to reproduce llama training dataset.

Hugo Touvron et al. 2023. Llama 2: Open foundation and fine-tuned chat models.

Jonathan Tow, Marco Bellagente, Dakota Mahan, and Carlos Riquelme. 2023. Stablelm 3b 4e1t.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2023. Zephyr: Direct distillation of lm alignment.

Yiren Wang, Fei Tian, Di He, Tao Qin, ChengXiang Zhai, and Tie-Yan Liu. 2019. Non-autoregressive machine translation with auxiliary regularization. *CoRR*, abs/1902.10245.

Lijun Wu, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2018. A study of reinforcement learning for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3612–3621, Brussels, Belgium. Association for Computational Linguistics.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. 2024a. WizardLM: Empowering large pre-trained language models to follow complex instructions. In *The Twelfth International Conference on Learning Representations*.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024b. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *Forty-first International Conference on Machine Learning*.

Michael Yeung, Evis Sala, Carola-Bibiane Schönlieb, and Leonardo Rundo. 2022. Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation. *Computerized Medical Imaging and Graphics*, 95:102026.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2024. MAmmoTH: Building math generalist models through hybrid instruction tuning. In *The Twelfth International Conference on Learning Representations*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, LILI YU, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. LIMA: Less is more for alignment. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Appendices

In this supplementary material, we provide additional details as follows:

- Appendix A: Loss Function Formulations
- Appendix B: Dataset Statistics
- Appendix C: Token Distribution
- Appendix D: Model Summary
- Appendix E: Extended Results
- Appendix F: Correlation between General purpose Metrics and ROSCOE Metrics
- Appendix G: Implementation Details
- Appendix H: Prompt Examples

A Loss Function Formulations

For the sake of simplicity, hereinafter we will consider the binary formulation. However, the loss formulations can be straightforwardly extended to the multi-class scenario.

Cross Entropy Loss Accuracy (AC) and Cross-Entropy Loss (CE) are defined as follows:

$$\text{AC} = \frac{1}{N}\sum_{i}^{N} 1(\hat{y}_i = y_i) \tag{1}$$

$$\text{CE}(p_t) = -\log(p_t) \tag{2}$$

where N is the total number of samples, $\hat{y}_i$ and y_i are the predicted and ground truth class for sample i, respectively, and p_t is the probability of the sample belonging to the positive class.

Focal Loss Focal Loss (FL) (Lin et al., 2017) can be defined as follows:

$$\text{FL}(p_t) = -(1-p_t)^{\gamma}\log(p_t) \tag{3}$$

where p_t is the probability of the sample belonging to the positive class while γ is the Focal suppression parameter.

Dice Loss Dice Score and Dice Loss (DL) (Milletari et al., 2016) are defined as follows:

$$\text{DS} = \frac{2|\hat{Y} \cap Y|}{|\hat{Y}| + |Y|} = \frac{2TP}{2TP + FP + FN} \tag{4}$$

$$\text{DL} = 1 - \frac{2\sum_i p_i y_i}{\sum_i p_i^2 + \sum_i y_i^2} \tag{5}$$

where $\hat{Y}$ and Y are the prediction and ground truth sets, TP, FP, FN are the numbers of true positives, false positives, and false negatives, respectively, p_i is the probability of the sample belonging to the positive class, and y_i is the ground truth label.

Self-Adjusting Dice Loss Self-Adjusting Dice Loss (SADL) (Li et al., 2020) can be expressed as follows:

$$\text{SADL} = 1 - \frac{2\sum_i (1-p_i)p_i y_i}{\sum_i (1-p_i)p_i + y_i} \tag{6}$$

where $(1-p_i)$ is the Focal component in Equation (3).

Generalized Dice Loss Generalized Dice Loss (GDL) (Sudre et al., 2017) can be expressed as follows:

$$\text{GDL} = 1 - \frac{2\sum_l w_l \sum_i p_{il} y_{il}}{\sum_l w_l \sum_i p_{il} + y_{il}} \tag{7}$$

where $w_l = 1/(\sum_i y_{il})^2$ for each class, while p_i and y_i have the same meanings as defined in Equation (5).

Lovász Loss Let $\hat{Y}$ and Y represent the prediction and ground truth sets, respectively. The Jaccard Index (or Intersection-over-Union, IoU) is defined as follows:

$$\text{IoU} = \frac{|\hat{Y} \cap Y|}{|\hat{Y} \cup Y|} = \frac{TP}{TP + FP + FN} \tag{8}$$

Lovász surrogate Loss (LL) (Berman et al., 2018) has the following expression:

$$\Delta_{J_1} = 1 - \frac{|\{\hat{Y} = 1\} \cap \{Y = 1\}|}{|\{\hat{Y} = 1\} \cup \{Y = 1\}|} \tag{9}$$

$$HL_i(x_i, y_i) = \max(0, 1 - x_i y_i) \tag{10}$$

$$\text{LL} = \overline{\Delta_{J_1}} HL(X, Y) \tag{11}$$

where Δ_{J_1} is the Jaccard loss, HL is the hinge loss, $x_i \in X$ is the prediction logit associated to sample i, $y_i \in Y$ with $y_i \in \{-1, 1\}$, and $\overline{\Delta_{J_1}}$ is the Lovász extension of the Jaccard loss.

B Dataset Statistics

- GSM8K[2] (Cobbe et al., 2021) is a dataset of 8.5K high-quality linguistically diverse grade

[2] https://huggingface.co/datasets/gsm8k

school Math Word Problems. The dataset was created to support answering questions on basic mathematical problems requiring multi-step reasoning. It has 7470 samples in the training set and 1320 in the test set. It is released under the MIT license.

- MathQA[3] (Amini et al., 2019) is a large-scale dataset of Math Word Problems enhancing the AQuA dataset (Ling et al., 2017) by providing fully-specified operational programs for each problem. It comprises 29800, 4480, and 2990 samples in the training, validation, and test sets, respectively. It is released under the Apache-2.0 license.

- OpenBookQA[4] (Mihaylov et al., 2018) contains questions that require multi-step reasoning, use of additional common and commonsense knowledge, and rich text comprehension. OpenBookQA is modeled after open-book exams for assessing human understanding of a subject. The training, validation, and test sets contain 4960, 500, and 500 samples, respectively. It is released under the Apache-2.0 license.

- HellaSwag[5] (Zellers et al., 2019) introduced a task of commonsense natural language inference, which consists in selecting the most appropriate conclusion for a sentence from a set of possibilities. It contains 39900 samples in the training set and 10000 in the validation set, which is employed as the test set since the actual test set does not have ground truth. It is released under the MIT license.

C Token Distribution

We report in Figure 3 the distribution of tokens across the datasets, highlighting the strong imbalance in tokens. Before the analysis, we excluded all special tokens (25) from the tokenizer. We plot the density against the token identifier in the log scale to better highlight peaks and differences.

D Model Summary

Table 7 summarizes the characteristics of the models used in this work: RedPajama-Incite-3B[6], StableLM-3B[7], RedPajama-Incite-7B[8], Falcon-7B[9], and Llama-2-7B[10]. For each of them, the following characteristics are reported: model name, number of parameters, license, availability of the pre-training datasets, and mean win rate according to HELM benchmark (Liang et al., 2022).

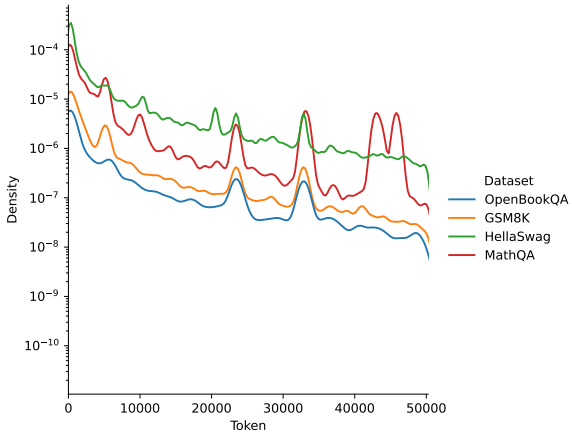

Figure 3: Kernel Density Estimation in log scale for token distributions in GSM8K, MathQA, OpenBookQA, and HellaSwag datasets.

D.1 Competitors

The competitors chosen are: MAmmoTH[11], WizardMath[12], WizardLM[13], Llemma[14], MetaMath[15], Mistral-7B[16], and GPT-4. We employed the settings and prompts suggested by the authors of the original papers.

MAmmoTH is released under the MIT license. Mistral is released under the Apache 2.0 license. The other models are released under the Llama 2 license.

E Extended Results

In the following, we report the extended results for the mathematical reasoning and question-

[3] https://huggingface.co/datasets/math_qa
[4] https://huggingface.co/datasets/openbookqa
[5] https://huggingface.co/datasets/Rowan/hellaswag
[6] https://huggingface.co/togethercomputer/RedPajama-INCITE-Base-3B-v1
[7] https://huggingface.co/stabilityai/stablelm-3b-4e1t
[8] https://huggingface.co/togethercomputer/RedPajama-INCITE-7B-Base
[9] https://huggingface.co/tiiuae/falcon-7b
[10] https://huggingface.co/meta-llama/Llama-2-7b-hf
[11] https://huggingface.co/TIGER-Lab/MAmmoTH-7B
[12] https://huggingface.co/TheBloke/WizardMath-7B-V1.1-GPTQ
[13] https://huggingface.co/TheBloke/wizardLM-7B-HF
[14] https://huggingface.co/EleutherAI/llemma_7b
[15] https://huggingface.co/meta-math/MetaMath-7B-V1.0
[16] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2

Model	# Parameters	License	Pre-Training Datasets	HELM Win Rate
RedPajama-Incite	3B	Apache 2.0	Public	0.311
StableLM	3B	CC BY-SA-4.0	Public	–
RedPajama-Incite	7B	Apache 2.0	Public	0.378
Falcon	7B	Apache 2.0	90% Public	0.378
Llama-2	7B	Llama-2	Public	0.607

Table 7: Model characteristics.

answering tasks.

E.1 Complete results on MWP

In Tables 9 and 10, we present the detailed performance of each model and loss function on MWP datasets. We use McNemar's test for exact match and t-tests (Dietterich, 1998) for other metrics to determine if differences are statistically significant. According to our metrics in GSM8K, Lovász provides the best mean performance across all models, except on Falcon, where Self-Adjusting Dice yields the best results. However, differences are not statistically significant, likely due to the model's limitations. On MathQA, Lovász achieves the best performance across most metrics, while for exact match, Focal performs best 2 out of 5 times. The results for ROSCOE metrics in Table 11 across both MWP datasets show that Lovász performs best in most metrics, as highlighted by the mean rank as well.

E.2 Complete results on Question Answering

In Table 12, we present the detailed performance of each model and loss function on closed-ended QA datasets. We perform McNemar's test (Dietterich, 1998) to assess whether differences compared to cross-entropy loss alone are statistically significant. In 6 cases, Lovász loss provides the best improvements, while in 4 cases, Focal loss obtains the best results. The main differences are seen when Lovász fails, whereas Focal still gets improvement. In the opposite case, the results are similar.

F Correlation between General purpose Metrics and ROSCOE Metrics

In Table 8, we report the Pearson's correlation analysis between Exact Match (EM), Precision (Prec), Recall (Rec), Dice Score (DS), Intersection-over-Union (IoU), Commutative Intersection-over-Union (C-IoU) and ROSCOE metrics, showing medium-high correlation values. Reasoning Alignment (RA) and Redundancy (RD) exhibit the strongest correlations with the general-purpose metrics. Common Sense Error (CSE) and Semantic Coverage Chain (SCC) demonstrate moderate correlation values. External Hallucination (EH) and Missing Steps (MS) show a moderate correlation as well.

	EM	IoU	Prec	Rec	DS	C-IoU
RA (SA)	0.1615	0.6582	0.6891	0.6076	0.6739	0.6698
EH (SA)	0.1425	0.6058	0.6186	0.5115	0.5919	0.6074
RD (SA)	0.1607	0.6781	0.6911	0.5674	0.6600	0.6828
CSE (SA)	0.1559	0.5583	0.5314	0.5741	0.5596	0.5608
MS (SA)	0.1744	0.6461	0.6138	0.6595	0.6463	0.6523
SCC (SS)	0.1345	0.5403	0.5501	0.5005	0.5484	0.5495

Table 8: Pearson's correlation between reasoning metrics (ROSCOE) and standard ones (EM, IoU, Prec, Rec, DS, C-IoU) over all samples.

G Implementation Details

Based on preliminary experiments, we set the language modeling loss mixing parameter to $\lambda = 0.6$. The Focal suppression parameter was set to $\gamma = 2$. The maximum learning rate was set to $1e-4$ for all datasets, except in GSM8K, for which it was set to $1e-5$.

We selected the model checkpoints according to the best validation loss. We train less than 1% of the total model parameters using LoRA. During training, the context size is chosen to include most samples without truncation according to 75% percentiles: 128 for GSM8K, MathQA, OpenBookQA, and 256 for HellaSwag. We employ gradient accumulation for context size 256.

We employed Transformers and PEFT libraries. Full requirements, versions, and losses' licenses are available in the code repository. For ROSCOE metrics evaluation, we employed the models suggested in the original paper: SimCSE[17] for sentence embedding, RoBERTa[18] as word embedding

[17] https://huggingface.co/facebook/roscoe-512-roberta-base

[18] https://huggingface.co/FacebookAI/

model, DeBERTa[19] as NLI model, RoBERTa[20] as grammar model, and GPT-2[21] as perplexity model.

We ran our experiments on a machine equipped with Intel® Core™ i9-10980XE CPU, 1 × NVIDIA® RTX A6000 48GB GPU, 128 GB of RAM running Ubuntu 22.04 LTS.

H Prompt Examples

We express the prompts to fine-tune the LLMs considered as follows:
Question: [Question Text] (Context: [Context text])
Answer: [Answer Text]
where *Context* is optional as not every dataset includes it. The answer format can be either a single letter corresponding to the answer for QA or a series of passages and a final answer for mathematical reasoning problems. In the latter case, we adhere to the format of GSM8K:
«[Formula]» ... #### [Final answer]
where each *Formula* comprises operators and operands, which can be numbers or symbols. This is done to better evaluate mathematical steps, which exhibit less ambiguity and adhere to stricter lexical rules than textual reasoning. In the following, we include some example prompts.

GSM8K *Question: John takes care of 10 dogs. Each dog takes .5 hours a day to walk and take care of their business. How many hours a week does he spend taking care of dogs?*
Answer: «10.5=5» «5*7=35» #### 35*

MathQA *Question: Sophia finished 2/3 of a book . she calculated that she finished 90 more pages than she has yet to read . how long is her book ?*
Answer: «divide(n0,n1)» «subtract(const_1,#0)» «divide(n2,#1)» #### 270

OpenBookQA *Question: Stars are*
A. warm lights that float
B. made out of nitrate
C. great balls of gas burning billions of miles away
D. lights in the sky
Context: a star is made of gases
Answer: C

HellaSwag *Question: A female chef in white uniform shows a stack of baking pans in a large kitchen presenting them. the pans*
A. contain egg yolks and baking soda.
B. are then sprinkled with brown sugar.
C. are placed in a strainer on the counter.
D. are filled with pastries and loaded into the oven.
Answer: D

roberta-base
[19] https://huggingface.co/MoritzLaurer/DeBERTa-v3-large-mnli-fever-anli-ling-wanli
[20] https://huggingface.co/cointegrated/roberta-large-cola-krishna2020
[21] https://huggingface.co/openai-community/gpt2-large

Model	Loss	EM	IoU	Prec	Rec	DS	C-IoU
RedPajama 3B	CE	9.33	11.03	14.66	15.51	14.69	14.76
	FL	9.55	11.46	15.23	16.16	15.33	15.21
	GDL	9.25	11.15	14.81	15.67	14.83	14.92
	LL	**11.45***	**12.52***	**16.66***	**17.17***	**16.52***	**16.53***
	SADL	10.16	11.76	15.80*	16.19	15.60	15.73*
StableLM 3B	CE	24.79	20.96	26.05	26.72	25.93	24.56
	FL	24.79	21.81*	27.36*	27.49	26.95*	25.51*
	GDL	24.87	21.01	26.11*	26.75	25.98	24.58
	LL	**28.66***	**24.02***	**29.42***	**30.38***	**29.38***	**28.15***
	SADL	26.99*	21.08	26.43	27.40	26.39	25.20
RedPajama 7B	CE	16.07	15.39	19.93	20.38	19.76	19.76
	FL	14.94	14.93	19.92	19.55	19.32	18.82
	GDL	13.19*	13.94*	18.27*	19.24*	18.33*	17.94*
	LL	**16.83**	**16.66***	**21.57***	**21.52**	**21.13***	**20.91***
	SADL	13.95*	14.94	19.32	20.41	19.44	18.85
Falcon 7B	CE	4.70	11.39	14.00	20.64	16.15	14.16
	FL	3.49	9.19*	11.25*	19.47	13.69*	11.92*
	GDL	4.40	11.16	13.65	20.85	15.98	13.98
	LL	5.00	11.59	13.93	22.09	16.47	14.08
	SADL	**5.08**	**12.04**	**14.37**	**23.70**	**17.18**	**15.00**
Llama-2 7B	CE	24.28	18.85	23.62	23.92	23.35	23.13
	FL	24.28	18.07	22.61	23.78	22.76	22.07
	GDL	23.29	18.47	23.26	23.64	23.01	22.07
	LL	**26.86***	**22.14***	**27.09***	**27.74***	**26.93***	**25.83***
	SADL	23.37	18.36	22.98	24.03	23.01	22.78

Table 9: Results on GSM8K dataset. * indicates values for which $p < 0.05$.

Model	Loss	EM	IoU	Prec	Rec	DS	C-IoU
RedPajama 3B	CE	**3.47**	30.26	34.20	35.32	34.07	30.29
	FL	2.79	**33.11***	**37.29***	37.87*	**36.88***	**33.16***
	GDL	2.45*	28.98*	32.96*	33.96*	32.72*	29.06*
	LL	2.83	32.83*	36.48*	**38.44***	36.69*	32.86*
	SADL	2.79	26.54*	30.35*	32.55*	30.49*	26.58*
StableLM 3B	CE	8.21	61.98	64.86	67.39	65.36	62.02
	FL	**10.06***	61.98*	65.43*	67.47*	65.66*	62.04*
	GDL	6.86	57.13*	60.16*	63.61*	61.03*	57.16*
	LL	7.50	**65.73***	**68.51***	**70.79***	**69.06***	**65.80***
	SADL	7.16	59.79*	62.85*	65.31*	63.33*	59.84*
RedPajama 7B	CE	7.16	40.35	44.32	45.01	43.98	40.41
	FL	**8.78***	43.12*	47.72*	48.28*	47.16*	43.17*
	GDL	7.05	41.21*	44.87*	45.98*	44.77*	41.27*
	LL	6.82	**46.34***	**49.87***	**51.27***	**49.92***	**46.41***
	SADL	6.10	32.41*	39.17	36.75*	36.79	32.48*
Falcon 7B	CE	5.24	11.34	13.80	21.72	15.93	11.44
	FL	5.84	10.93*	12.98*	24.59*	15.77*	11.00*
	GDL	5.69	11.07*	13.21*	22.98*	15.63*	11.14*
	LL	5.35	**12.77**	**15.00***	**26.07***	**17.67***	**12.87**
	SADL	**5.99**	10.57*	12.62*	21.50*	14.84*	10.63*
Llama-2 7B	CE	1.51	39.69	44.34	45.45	43.98	39.75
	FL	0.15*	19.51*	22.29*	30.48*	24.43*	19.60*
	GDL	**3.17***	43.12*	45.56	57.74*	48.87*	43.16*
	LL	1.28	**58.56***	**61.00***	**66.16***	**62.28***	**58.62***
	SADL	0.38*	41.57*	43.45	58.87*	47.77*	41.62*

Table 10: Results on MathQA dataset. * indicates values for which $p < 0.05$.

	CE	FL	GDL	LL	SADL
Faithfulness	81.96	81.97	81.98	**82.21**	81.96
Informativeness Step	80.61	81.09	**81.11**	80.82	81.10
Faithfulness WW	91.84	92.61	**92.78**	91.55	92.77
Informativeness Chain	90.63	90.40	90.50	**90.79**	90.41
Repetition Word	12.59	13.58	9.80	**15.67**	10.91
Repetition Step	14.44	16.02	12.30	**17.40**	13.30
Reasoning Alignment	92.47	92.37	**92.67**	92.61	92.60
External Hallucination	97.59	97.60	97.57	**97.70**	97.58
Redundancy	88.71	88.60	88.69	**89.06**	88.62
Common Sense Error	97.91	97.87	**97.96**	**97.96**	97.93
Missing Step	89.47	89.47	**89.89**	89.82	89.74
Semantic Coverage Step	98.14	98.25	98.31	**98.32**	98.27
Semantic Coverage Chain	96.21	96.17	**96.36**	96.35	96.30
Discourse Representation	42.71	42.73	41.50	**45.68**	40.95
Perplexity Step	0.28	0.27	**0.28**	0.26	0.27
Coherence Step vs Step	16.41	17.76	14.21	**19.00**	14.94
Perplexity Chain	6.08	6.42	6.74	5.49	**6.84**
Perplexity Step Max	0.14	0.13	0.14	0.14	**0.15**
Grammar Step	94.27	94.18	94.12	**94.28**	94.18
Grammar Step Max	90.32	90.02	89.95	**90.34**	90.00
Mean Rank	3.20	3.45	2.80	**1.95**	3.20

Table 11: Results using ROSCOE metrics aggregated across models and datasets.

Model	Loss	HellaSwag	OpenBookQA
RedPajama 3B	CE	25.26	66.60
	FL	**45.91***	**78.60***
	GDL	25.39	63.80
	LL	26.05	77.20*
	SADL	25.79*	67.00
StableLM 3B	CE	79.69	84.00
	FL	**85.69***	85.40
	GDL	80.00	82.80
	LL	82.97*	**87.20***
	SADL	80.49*	82.40
RedPajama 7B	CE	25.16	74.80
	FL	**73.29***	81.60*
	GDL	25.04	75.80
	LL	25.08	**83.80***
	SADL	25.10	76.60
Falcon 7B	CE	24.59	69.20
	FL	68.51*	77.20*
	GDL	24.94	69.20
	LL	**70.72***	**79.00***
	SADL	26.67*	55.00*
Llama-2 7B	CE	82.12	83.40
	FL	85.03*	81.60
	GDL	81.58	83.80
	LL	**85.60***	**86.80***
	SADL	51.10*	56.00*

Table 12: Results on Question Answering datasets. * indicates values for which $p < 0.05$.