AUTONOMOUS DATA SELECTION WITH LANGUAGE MODELS FOR MATHEMATICAL TEXTS

Yifan Zhang[1*] **Yifan Luo**[2,4*] **Yang Yuan**[1,2,3†] **Andrew Chi-Chih Yao**[1,2,3†]
[1]IIIS, Tsinghua University
[2]Shanghai Qizhi Institute
[3]Shanghai Artificial Intelligence Laboratory
[4]SCSS, Beijing University of Posts and Telecommunications
zhangyif21@mails.tsinghua.edu.cn,
{yuanyang,andrewcyao}@tsinghua.edu.cn

ABSTRACT

To improve language models' proficiency in mathematical reasoning via continual pretraining, we introduce a novel strategy that leverages base language models for autonomous data selection. Departing from conventional supervised fine-tuning or trained classifiers with human-annotated data, our approach Autonomous Data Selection (AutoDS) utilizes meta-prompted language models as zero-shot verifiers to evaluate and select high-quality mathematical content autonomously. To demonstrate the efficacy of our method, we continuously pretrained a 7B-parameter language model on our curated dataset, achieving substantial improvements in downstream performance on the MATH, GSM8K, and BIG-Bench Hard (BBH) tasks with a token amount reduced by orders of magnitude compared to previous continual pretraining works. Our method showcases a 2 times increase in pretraining token efficiency compared to state-of-the-art baselines, underscoring the potential of our approach in enhancing models' mathematical reasoning capabilities. The AutoMathText dataset is available at https://huggingface.co/datasets/math-ai/AutoMathText[‡].

1 INTRODUCTION

In the field of language modeling research (Devlin et al., 2018; Radford et al., 2018; 2019; Brown et al., 2020; OpenAI, 2023; Anil et al., 2023), the incorporation of domain-specific knowledge emerges as a crucial area for exploration (Lewkowycz et al., 2022; Azerbayev et al., 2023b). This is particularly important in the realm of mathematical reasoning, where the development and curation of specialized datasets for pretraining and finetuning represent a critical need and a challenge (Hendrycks et al., 2021; Paster et al., 2023; Wang et al., 2023). The drive toward creating language models proficient in complex mathematical reasoning underscores the importance of high-quality, domain-specific datasets. However, the mathematical field faces a scarcity of such resources, highlighting the need for innovative solutions to cultivate models with deep understanding and problem-solving skills.

Recent endeavors, such as those by Gunasekar et al. (2023) and Li et al. (2023), have made significant strides in addressing this challenge. They demonstrated the potential of leveraging GPT-4 to assess the educational value of code data within the Stack dataset (Kocetkov et al., 2022), employing model-generated annotations to train a random forest classifier for quality prediction. These studies mark a pivotal step toward enhancing the quality of data for model training. Nonetheless, they can only assign discrete labels to the data points, e.g., good or bad, instead of assigning continuous real scores, e.g., a data point of educational value 0.95 vs a data point of value 0.001.

As we will demonstrate later, computing real-valued scores for training data can significantly improve the pretraining token efficiency because the model can focus on the most informative data points,

[*]Equal contribution.
[†]Corresponding authors.
[‡]The code is available at https://github.com/yifanzhang-pro/AutoMathText.

where “informative” is defined by a scoring threshold. However, generating scores can be difficult for large language models (LLMs), as it has been observed that LLMs are not good at accurately generating numbers or sampling from complex distributions (Hopkins et al., 2023; Hu et al., 2023). Inspired by the innovative DPO method (Rafailov et al., 2023), we propose leveraging the logits of specific tokens to directly formulate a quantitative score function, circumventing the need for extensive data labeling or classifier training.

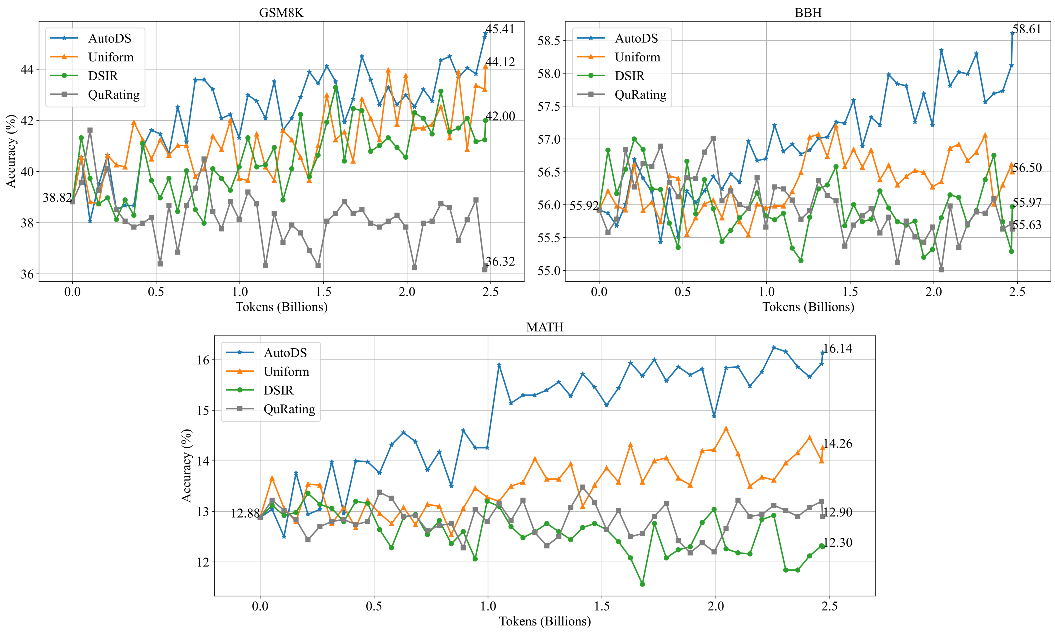

Figure 1: Visualization of performances of continual pretrained models with different data selection methods on GSM8K (Hendrycks et al., 2021), BIG-Bench Hard (BBH) (Suzgun et al., 2022) and MATH (Hendrycks et al., 2021) tasks.

In this work, we introduce a strategy that utilizes the intrinsic capabilities of base language models, equipped with zero-shot meta-prompts, to autonomously evaluate the mathematical quality and educational value of content. Our score function offers a more nuanced and granular analysis, unlike previous methods that primarily focused on binary classification (Li et al., 2023; Paster et al., 2023). This enables a refined and sophisticated training strategy that extends beyond the limitations of binary filtering.

The core of our contribution lies in the autonomous content evaluation without the necessity for alignment with human-labeled scores through Supervised Fine-Tuning (SFT), Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022), or Direct Preference Optimization (DPO) (Rafailov et al., 2023). By employing a softmax function over logits for ‘YES’ and ‘NO’ tokens, our method autonomously assesses content relevance and value. This facilitates an active learning process where the model customizes its learning journey by querying the educational merit of materials. This approach signifies an attempt towards the realization of autonomous learning systems that are dynamic, proactive, and capable of self-directed evaluation and learning, especially in specialized fields like mathematics.

Our contributions can be listed as three-fold:

- We showcase the efficacy of leveraging base language models with meta-prompts for zero-shot verification using a straightforward score function derived from logits. Our method, Autonomous Data Selection (AutoDS) advances beyond traditional alignment strategies such as SFT and RLHF without the reliance on human-annotated data, facilitating autonomous content evaluation.
- We address the shortage of labeled high-quality mathematical training resources by introducing the open-source AutoMathText dataset. This comprehensive dataset is designed to enrich AI

model training with mathematical content, thereby enhancing their performance in math-intensive tasks.

- Through empirical evidence, we demonstrate the effectiveness of our methodology by continuously pretrain a 7B parameter Mistral language model on the AutoMathText dataset. Our results highlight substantial improvements in downstream performance on the MATH (Hendrycks et al., 2021), GSM8K (Cobbe et al., 2021), and BIG-Bench Hard (BBH) tasks (Suzgun et al., 2022) with 2 times pretraining token efficiency, underscoring the practical benefits of our approach in mathematical reasoning tasks.

2 LANGUAGE MODELS AS ZERO-SHOT VERIFIERS

The proliferation of language models has introduced unprecedented opportunities for advancing AI systems capable of intricate reasoning and decision-making (Wei et al., 2022; Bubeck et al., 2023). In this context, our work explores the frontier of employing base language models as zero-shot verifiers, a concept that diverges from traditional few-shot learning paradigms (Brown et al., 2020) by eliminating the need for task-specific fine-tuning or example-based prompting (Reynolds & McDonell, 2021; Kojima et al., 2022; Zhang et al., 2023b). Our methodology embraces the zero-shot approach to leverage the inherent capabilities of language models, thereby enabling a direct assessment of textual content's relevance and educational value in the domain of mathematics without prior alignment with human-generated labels.

Central to our approach AutoDS is the formulation of a scoring function, as delineated in Equation (1), which quantitatively evaluates the language model's inclination towards affirming or negating the mathematical content and educational merit of a given piece of content. This function operates on the logits associated with 'YES' and 'NO' responses to meta-prompts, offering a nuanced mechanism for content evaluation:

$$\text{LM-Score}(\cdot) = \frac{\exp(\text{logit}(\text{`YES'}))}{\exp(\text{logit}(\text{`YES'})) + \exp(\text{logit}(\text{`NO'}))}. \tag{1}$$

This scoring function represents a novel integration of language models' prediction capabilities into an autonomous evaluation framework, bypassing the limitations associated with traditional supervised learning techniques. Our approach forgoes the conventional reliance on manually labeled datasets or classifier training, instead offering a direct and nuanced assessment of content across varied mathematical sources, as exemplified in Figures 3 and 7. Figure 2 demonstrates the meta-prompt designed for autonomous data selection, illustrating how language models can evaluate the mathematical and educational value of content from diverse sources such as Common Crawl, arXiv, and GitHub (see Figure 8 and 9). Our use of meta-prompts not only serves as in-context alignment for base language models but also ensures that the language models operate within a specifically tailored syntax, enhancing their ability to produce type-safe foreseeable responses. Notice that the '<system>' tags are directly using plain text instead of special tokens for ease of implementation without modifying the tokenizers. Responses from the model are constrained to four possibilities, thereby allowing for a refined selection process tailored to educational content in mathematics.

Leveraging the capacity for handling multiple queries within a single prompt, our methodology interprets the LM score as a pseudo-probability. This interpretation facilitates a layered assessment by aggregating the scores of individual questions. In our framework, the language model is tasked with addressing two queries simultaneously, and we derive the composite LM-Score for these inquiries utilizing Equation (2). In subsequent discussions, we refer to this aggregated measure simply as the LM-Score. This approach emphasizes the redundancy of collecting annotated data for alignment techniques like Supervised Fine-Tuning (SFT) or Reinforcement Learning from Human Feedback (RLHF), proposing a more streamlined, zero-shot in-context alignment strategy. This refined strategy not only simplifies the evaluation process but also enhances the efficiency and scalability of our AutoDS method.

$$\text{LM-Score}(Q_1, Q_2) = \text{LM-Score}(Q_1) \cdot \text{LM-Score}(Q_2). \tag{2}$$

Importantly, the utilization of base language models equipped with meta-prompts is instrumental in our approach, offering a highly efficient pathway for continual pretraining and active life-long

"
<system>
You are ChatGPT, equipped with extensive expertise in mathematics and coding, and skilled in complex reasoning and problem-solving. In the following task, I will present a text excerpt from a website. Your role is to evaluate whether this text exhibits mathematical intelligence and if it is suitable for educational purposes in mathematics. Please respond with only YES or NO
</system>

User: {
"url": "{url}",
"text": "{text}"
}
1. Does the text exhibit elements of mathematical intelligence? Respond with YES or NO
2. Is the text suitable for educational purposes for YOURSELF in the field of mathematics? Respond with YES or NO
"
Assistant: 1.

Figure 2: Illustration of a zero-shot meta-prompt designed for the AutoDS method.

learning. Through the strategic use of meta-prompts, we can tap into the innate instruction-following capabilities of these models, bypassing the need for traditional alignment mechanisms. This intrinsic property allows for the direct application of a model's latest checkpoint to autonomously determine the suitability of data for subsequent pretraining epochs. Such a method not only streamlines the process of data curation but also ensures that the model remains dynamically attuned to the evolving landscape of mathematical content, thereby enhancing its learning trajectory and adaptability over time. This underscores the transformative potential of our approach in leveraging the existing competencies of language models for autonomous data evaluation and selection, setting a new precedent for the development of self-evolving AI systems specialized in the domain of mathematics.

Moreover, our approach deliberately avoids SFT or RLHF to anticipate and leverage the evolving superiority of language models over human evaluative capabilities, especially in domains requiring specialized knowledge like mathematics. This decision is substantiated by the examples depicted in Figures 3 and 7, which highlight the potential limitations of trained classifier-based and human-led content evaluation. OpenWebMath (Paster et al., 2023) trained a model to predict the probability a document is mathematical, which turns out not to be very satisfying (see Figure 3).

Language models, free from human biases and constraints, present a scalable and objective mechanism for content assessment, as humans may be seen as weak supervisors compared to language models themselves (Burns et al., 2023). Our methodology advocates for autonomous supervision through direct engagement by eliciting language models. This paradigm shift towards self-supervised evaluation and selection paves the way for the next generation of AI systems, characterized by their autonomous learning and adaptability in specialized knowledge domains.

3 Autonomous Data Selection with Language Models for Mathematical Texts

Our study leverages three primary data sources: Common Crawl (specifically, the OpenWebMath subset (Paster et al., 2023)), arXiv (via the RedPajama dataset (Computer, 2023)), and GitHub (the Stack dataset (Kocetkov et al., 2022; Azerbayev et al., 2023b)). These sources were chosen for their rich mathematical content, spanning a broad spectrum of complexity and formats.

Experiment Details. We employ the Qwen-72B base language model (Bai et al., 2023), notable for its MMLU score of 77.4, to process our datasets. Specifically, we utilize:

1. 6.32M documents from the OpenWebMath dataset (Paster et al., 2023), a curated subset of Common Crawl;
2. 1.54M documents from the arXiv subset of the RedPajama dataset (Computer, 2023);

"Commutative Property Of Addition. If A is an $n \times m$ matrix and O is a $m \times k$ zero-matrix, then we have: AO = O. Note that AO is the $n \times k$ zero-matrix. ..."
[LM-Score (Q_1, Q_2): 0.946]
[OWMath Classifier Score: 0.767]

"Inequality involving sums with binomial coefficient I am trying to show upper- and lower-bounds on $\frac{1}{2^n}\sum_{i=0}^{n}\binom{n}{i}\min(i, n-i)$ (where $n \geq 1$) to show that it grows as $\Theta(n)$. The upper-bound is easy to get since $\min(i, n-i) \leq i$ for $i \in \{0, \ldots n\}$ so that $\frac{1}{2^n}\sum_{i=0}^{n}\binom{n}{i}\min(i, n-i) \leq \frac{1}{2^n}\sum_{i=0}^{n}\binom{n}{i} i = \frac{n}{2}$. ..." [LM-Score ($Q_1$, Q_2): 0.931]
[OWMath Classifier Score: 0.999]

"The radius of convergence is half the length of the interval of convergence. We noticed that, at least in the case of the geometric series, there was an interval in which it converged, but it didn't converge at the endpoints. Show that the following alternating harmonic series converges: Series of Both Positive and Negative Terms Theorem: Convergence of Absolute Values Implies Convergence If $\sum |a_n|$ converges, then so does $\sum a_n$. Let $f : [1, \infty) \to \mathbb{R}_+$ be a non-negative ... "
[LM-Score (Q_1, Q_2): 0.923]
[OWMath Classifier Score: 0.906]

"# User talk:173.79.37.192 ## March 2009 Welcome to Wikipedia. Although everyone is welcome to make constructive contributions to Wikipedia, at least one of your recent edits, such as the one you made to Reaction time, did not appear to be constructive and has been reverted. Please use the sandbox for any test edits you would like to make, and read the welcome page to learn more about contributing constructively to this encyclopedia. Thank you. Hotcrocodile (talk) 01:33, 11 March 2009 (UTC) If this is a shared IP address, and you didn't make any unconstructive edits, consider creating an account for yourself so you can avoid further irrelevant warnings. ## NAYLA MATTHW [1] [[Media:Example.oggfhf... "
[LM-Score (Q_1, Q_2): 1.58×10^{-5}]
[OWMath Classifier Score: 0.612]

" I've just had one recent comment flag declined on a noisy comment. This comment was a reply to a deleted '+1' comment and said simply: @FrankL Thanks! "
[LM-Score (Q_1, Q_2): 1.21×10^{-5}]
[OWMath Classifier Score: 0.830]

Figure 3: Several examples on selecting web texts. The first example in the left column is from 'track-it.nz', while the second one in the left column is from 'math.stackexchange.com', and the third one in the left column is from 'bwni.pw'. In the right column, the first example is from 'wikipedia.org', and the second one is from ''math.stackexchange.com'. The trained classifier (denoted as OWMath Classifier) used in OpenWebMath (Paster et al., 2023) may mainly focus on how many latex symbols, $ and digits exist in the text, and the examples in the right column show that it may not be very effective.

3. 3.40M documents from the Algebraic Stack dataset (Azerbayev et al., 2023b), a specialized subset of the Stack dataset.

This selection encompassing over 200GB of data, while not exhaustive, serves as a representative demonstration, prioritizing cost-effectiveness and coverage. Our computational setup includes A100-80G and A800-80G GPUs, employing the vLLM inference framework (Kwon et al., 2023) for efficient language model inference. Processing the combined 11.26M documents required approximately 750 hours on 4 A100-80G GPUs, translating to 3000 GPU hours in total. Contrastingly, manual annotation of this dataset by experts familiar with undergraduate-level and beyond mathematical content would cost upwards of $10 million, assuming a rate of $1 per document. Our method significantly reduces this cost to approximately $10,000 (the cost is estimated by using Azure's machine learning service at $3.4 per A100 GPU hour).

3.1 Visualization of Data Composition

The visualization of data composition is essential to discern the quality and diversity of the web subset of our datasets. Figure 4 displays a tree map detailing the Top_{30} domains by LM-Score (Q_1, Q_2) ranges from 0.50 to 1.00 and 0.75 to 1.00, respectively. This representation not only spotlights the disparity in quality across different sources but also reveals the high-quality nature of data from StackExchange. This domain stands out, showcasing a considerable volume of content that demonstrates superior quality, yet a substantial portion of this data remains unexplored in existing literature (Wang et al., 2023; Liu et al., 2024), signifying a valuable opportunity for further investigation.

Delving deeper, Figure 5 offers a granular view of the LM-Score distribution across the Top_{10} domains. It is apparent that StackExchange, mathhelpforum.com, and physicsforums.com are leading in terms of high-quality content, with the highest proportions of scores within the 0.75 to 1.00 range. This detailed breakdown elucidates the domains where our autonomous data selection method is particularly effective, guiding the strategic direction for subsequent data preprocessing and model training efforts.

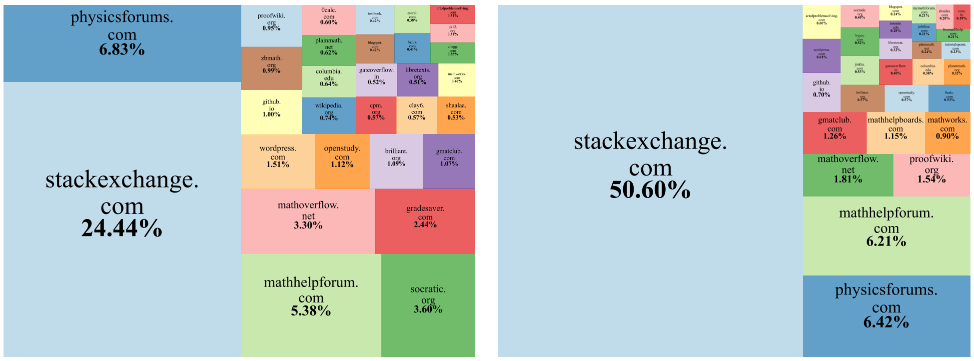

Figure 4: Data composition visualization for the Top_{30} domains, with LM-Score ranges highlighting content quality. The left one's LM-Scores are in the range 0.50-1.00, while the right one's LM-Scores are in the range 0.75-1.00.

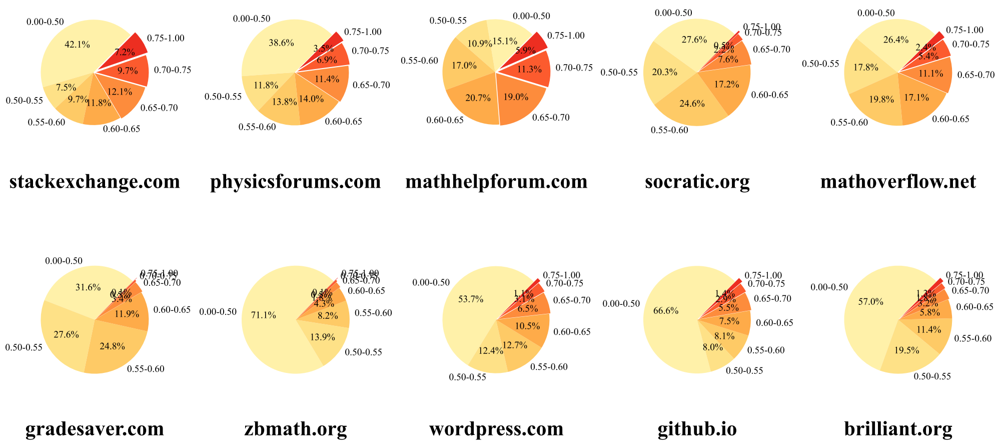

Figure 5: Visualization of LM-Score distribution within the Top_{10} domain occurrences, demonstrating the content quality and variety of different domains.

4 EXPERIMENTS

In this section, we want to test the effectiveness of the AutoDS method in enhancing the mathematical reasoning capabilities of language models. To this end, we continually pretrained a 7B-parameter Mistral language model (Jiang et al., 2023) showcasing the efficiency of our data selection method. Contrasting with the extensive 200B-token training performed by Llemma (Azerbayev et al., 2023b), we utilized merely less than 1.5% of that amount (less than 3B tokens), thereby emphasizing the potential of our data-efficient training approach. Our experiments include baselines employing uniform sampling, DSIR (Xie et al., 2023b), Qurating (Wettig et al., 2024), and our AutoDS method leveraging LM-Score-based selection. Token counts were balanced among different types of training data to ensure comparability.

4.1 Experiments on Continual Pretraining

Experiment details. Utilizing the LLaMA-Factory (hiyouga, 2023), we perform the continual pretraining of the Mistral-7B-v0.1 model with three epochs, using a cosine learning rate schedule with a 3% warm-up period and a peak learning rate of 5e-6. The DeepSpeed framework (Rajbhandari et al., 2020) with ZeRO-2 Stage optimization facilitates our training acceleration. The models are continual-pretrained on a node comprising 8xA800 GPUs. We use a micro-batch size of 8 and gradient accumulation of 4 to achieve the total batch size of 256. We first utilize the selected data from the web subset with the highest quality for a preliminary evaluation.

Evaluation results. Our evaluation protocol adheres to the standard eval harness framework (Gao et al., 2023a), consistent with the Huggingface Leaderboard's protocol [§]. The results, as detailed in the tables below, illuminate the efficacy of our AutoDS dataset in enhancing the model's performance.

Table 1: MATH test accuracy post continual pretraining.

LM-Score	Type	# Tokens (M)	Accuracy (%)
-	Baseline (w/o pretraining)	0	12.88
-	OpenWebMath	328.9	10.50
0.75-1.00	**AutoDS**	328.9	**13.68**

Table 2: MATH test accuracy after fine-tuning on MetaMathQA (Yu et al., 2023). Notice that the baseline accuracy is reproduced by ourselves for a fair comparison.

LM-Score	Type	# Tokens (M)	Accuracy (%)
-	Baseline (w/o pretraining)	0	27.20
-	OpenWebMath	328.9	26.98
0.75-1.00	**AutoDS**	328.9	**28.06**

In Table 1, we compare the MATH test accuracy of models after continual pretraining. The auto-selected data consistently outperforms its uniform counterpart, achieving higher accuracy percentages. Notice that the uniformly sampled data from the OpenWebMath dataset have already been filtered using OpenWebMath's rule-based filter and trained classifier. This enhancement in performance highlights the strategic advantage of using high-quality, domain-specific data for continual model pretraining. Table 2 further examines the MATH test accuracy after supervised fine-tuning (SFT) on the MetaMathQA dataset. In this SFT setting, the auto-selected data models again exhibit superior accuracy, affirming the robustness of our pretraining approach. These results underscore the AutoDS dataset's ability to enhance model performance and as a foundation for subsequent fine-tuning processes.

4.2 Comparison with Baselines

In this subsection, we conduct experiments at a larger scale to comprehensively evaluate different data selection methods. Specifically, we continually pretrained Mistral-7B models for one epoch with approximately 2.5 billion tokens.

Experiment details. In this experiment, we use a constant learning rate of 1e-6 as the default for all methods for fair comparisons. For the OpenWebMath dataset (Paster et al., 2023), we use uniform sampling as a baseline, notice that the OpenWebMath has already been selected using trained classifiers. The AutoDS method's data selection ranged from LM-Scores of 0.6 to 1.0 within the web subset. For the DSIR method, it requires a target dataset to calculate the KL divergence between the source dataset and the target dataset. We use the Pile (Gao et al., 2020) dataset's wiki validation set as the target dataset. For the Qurating method (Wettig et al., 2024), we directly utilized the QuratedPajama dataset, selecting data based on top-k scores of educational value.

Experimental results. In Figure 1, the checkpoint evaluation every 100 steps, approximately 52 million tokens. From Figure 1 and Table 3, models pretrained with the data selected using the AutoDS method consistently show superior performance across a diverse set of complex reasoning tasks

[§] `https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard`.

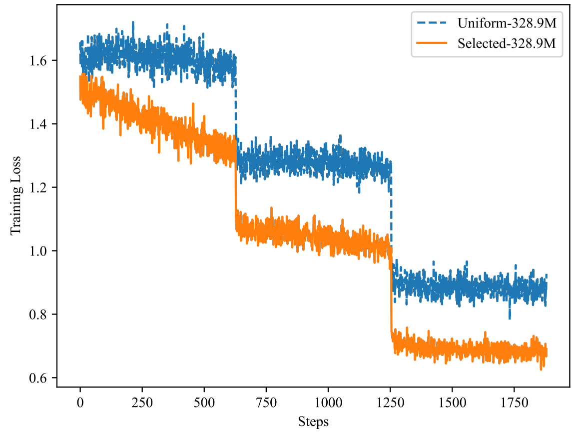

Figure 6: Comparative training loss trajectories for OpenWebMath (Uniform) and AutoMathText (Selected) data with size of 328.9M tokens, showcasing a substantial reduction in the training loss for Auto-selected AutoMathText data .

including MATH (Hendrycks et al., 2021), GSM8K (Cobbe et al., 2021), and BIG-Bench Hard (Suzgun et al., 2022), highlighting the method's robustness and the AutoDS dataset's effectiveness in enhancing models' reasoning capabilities. Notably, on the MATH dataset, AutoDS shows 2.36 times pretraining token efficiency compared to the OpenWebMath uniform sampling baseline (14.26%), achieving 16.14% accuracy with only 2.5B tokens for continual pretraining.

Beyond complex reasoning, we extended our evaluation to assess how well the models adapted to other cognitive domains, such as commonsense reasoning, world knowledge, and reading comprehension. Table 4 encapsulates this multi-faceted performance evaluation. It's noteworthy that while AutoDS did not universally top all categories, its overall average performance across diverse tasks (including all tasks shown in Table 3 and Table 4) shows the superiority of our method compared to other data selection methods. These outcomes strongly advocate for the AutoDS approach's potential to advance language models in mathematical reasoning and beyond.

Table 3: Comparison of continual pretrained models using different data selection methods on complex reasoning tasks, showcasing the notable superiority of the AutoDS method.

Selection Method	**MATH (5-shot)**	**GSM8K (5-shot)**	**BIG-Bench Hard (3-shot)**
– (Mistral-7B Base)	12.88	38.82	55.92
Uniform (OpenWebMath)	14.26	44.12	56.50
DSIR	12.30	42.00	55.97
QuRating	12.90	36.32	55.63
AutoDS	**16.14**	**45.41**	**58.61**

Table 4: Comprehensive comparison of continual pretrained models across diverse reasoning and comprehension tasks. The table is divided into three major sections: commonsense reasoning, world knowledge, and reading comprehension[§].

Selection Method	**H.S. (10)**	**PIQA (6)**	**W.G. (15)**	**NQ (5)**	**$\text{MMLU}_{\text{STEM}}$ (5)**	**ARC-E (25)**	**ARC-C (25)**	**SciQ (2)**	**LogiQA (2)**	**BoolQ (0)**	**Average**
– (Mistral-7B Base)	62.82	82.10	81.22	29.81	52.39	84.68	57.25	97.40	30.26	83.58	59.16
Uniform (OpenWebMath)	62.21	82.21	80.19	29.17	52.17	84.18	56.66	97.20	31.03	83.82	59.52
DSIR	63.10	81.94	81.37	29.22	52.62	84.72	57.25	97.30	30.26	73.76	58.59
QuRating	62.64	81.99	80.11	28.89	52.01	85.48	57.76	97.30	31.18	82.81	58.85
AutoDS	62.72	82.21	80.03	29.06	52.30	84.18	55.20	96.80	31.03	83.12	**59.76**

[§]Herein, H.S. denotes HellaSwag and W.G. signifies WinoGrande, and numbers with parenthetical reflecting few-shot example counts.

5 Related Work

Mathematical datasets and language models. The emergence of chain-of-thought prompting methodologies (Radford et al., 2019; Wei et al., 2022; Wang et al., 2022; Fu et al., 2022; Gao et al., 2023b; Yao et al., 2023; Zhang et al., 2023a; Gou et al., 2023) has been instrumental in harnessing and enhancing the reasoning capabilities inherent within language models. Our research, however, distinctly concentrates on the domain of continual pretraining with a focus on mathematical datasets. The creation of mathematical datasets has been critical in propelling AI models' proficiency in mathematical comprehension and reasoning. Foundational contributions, such as the AMPS dataset by Hendrycks et al. (2021) and the Proof-Pile dataset by Azerbayev et al. (2023a), have provided capstones for models to systematically tackle mathematical problems and proofs. The Llemma model (Azerbayev et al., 2023b) builds upon this foundation, dedicating its efforts to the continual pretraining of language models with mathematical data especially the OpenWebMath dataset (Paster et al., 2023), aiming to refine their complex reasoning skills further. Nevertheless, the meticulous selection of mathematical data is still an area fraught with challenges.

Data selection in language modeling. The landscape of data selection in language modeling has seen a variety of approaches aimed at refining the quality and relevance of training data. Techniques have ranged from employing binary classifiers used by GPT-3 (Brown et al., 2020) and PaLM (Chowdhery et al., 2023) to filter web data towards more formal sources like Wikipedia and books, to more nuanced strategies that consider the difficulty or domain-specificity of the data. For example, the Minerva model (Lewkowycz et al., 2022) used rule-based filtering for mathematical content, while DSIR (Xie et al., 2023b) applied importance resampling to align the data distribution with a target domain. Furthermore, DoReMi (Xie et al., 2023a) introduces a novel angle, optimizing domain weights with a proxy model to minimize worst-case excess loss across domains. However, the low inherent perplexity (entropy) in math-related and code-related corpora suggests that DoReMi might not be optimally suited for enhancing mathematical pretraining. Recently, Gunasekar et al. (2023); Li et al. (2023) demonstrated the utility of GPT-4 in annotating data quality for the Stack dataset (Kocetkov et al., 2022), subsequently using a random forest model for classification based on these annotations. Wettig et al. (2024) propose to train a reward model called Qurating for data selecting. Our work diverges from previous approaches by introducing a fully autonomous data selection method that leverages the intrinsic capabilities of language models without the need for human-generated (and AI-generated) annotations or external trained classifiers.

Data selection across various domains. The strategy of data selection transcends NLP tasks, extending its utility to a variety of domains, including vision and general domain adaptation. The Moore-Lewis technique, as introduced by Moore & Lewis (2010) and further refined by Axelrod (2017), exemplifies this approach by employing the cross-entropy differential between n-gram language models (LMs) tailored to specific targets and general corpora. Similarly, discrepancies in feature space and n-gram distributions have been effectively leveraged for data selection in domain adaptation scenarios, as evidenced by the work of Jiang & Zhai (2007), Liu et al. (2019), and Ruder & Plank (2017). Moreover, the significance of strategic data selection is equally acknowledged within the realm of computer vision, where methodologies aimed at optimizing training datasets have demonstrated substantial benefits. Notable contributions in this area include the pioneering curriculum learning framework by Bengio et al. (2009), the exploration of submodularity for efficient data selection by Wei et al. (2015), and recent advancements in prioritized data selection techniques by Coleman et al. (2019) and Mindermann et al. (2022).

6 Conclusion

Our method leverages the inherent self-evaluation and active learning capabilities of language models significantly improving the quality and relevance of training data in intricate and specialized fields like mathematics. This research opens the door to further investigations into autonomous data curation and model training techniques, heralding a new era in AI's capacity for understanding, reasoning, and innovation within specialized domains.

ETHIC STATEMENT

This study, aimed at enhancing the capabilities of language models through autonomous data selection and continual pretraining, presents insightful implications for the field of AI research, particularly in the training and development of language models with specialized knowledge. The deployment of autonomous systems for the selection of training data introduces considerations of transparency, fairness, and accountability within the AI development process. By reducing reliance on human-labeled data, our method shifts the responsibility for content evaluation to the AI itself, raising important questions about the model's decision-making processes. Ensuring these processes are transparent and free from biases is essential to prevent the perpetuation of existing inequalities or the introduction of new biases into AI systems.

REFERENCES

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023. 1

Amittai Axelrod. Cynical selection of language model training data. *arXiv preprint arXiv:1709.02279*, 2017. 9

Zhangir Azerbayev, Bartosz Piotrowski, Hailey Schoelkopf, Edward W Ayers, Dragomir Radev, and Jeremy Avigad. Proofnet: Autoformalizing and formally proving undergraduate-level mathematics. *arXiv preprint arXiv:2302.12433*, 2023a. 9

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen McAleer, Albert Q Jiang, Jia Deng, Stella Biderman, and Sean Welleck. Llemma: An open language model for mathematics. *arXiv preprint arXiv:2310.10631*, 2023b. 1, 4, 5, 6, 9

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 4

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009. 9

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1, 3, 9

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023. 3

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, et al. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. *arXiv preprint arXiv:2312.09390*, 2023. 4

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023. 9

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. 3, 8

Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. Selection via proxy: Efficient data selection for deep learning. *arXiv preprint arXiv:1906.11829*, 2019. 9

Together Computer. Redpajama: An open source recipe to reproduce llama training dataset, 2023. URL `https://github.com/togethercomputer/RedPajama-Data`. 4

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1

Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. Complexity-based prompting for multi-step reasoning. *arXiv preprint arXiv:2210.00720*, 2022. 9

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020. 7

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 12 2023a. URL `https://zenodo.org/records/10256836`. 7

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: Program-aided language models. In *International Conference on Machine Learning*, pp. 10764–10799. PMLR, 2023b. 9

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yujiu Yang, Minlie Huang, Nan Duan, Weizhu Chen, et al. Tora: A tool-integrated reasoning agent for mathematical problem solving. *arXiv preprint arXiv:2309.17452*, 2023. 9

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023. 1, 9

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021. 1, 2, 3, 8, 9

hiyouga. Llama factory. `https://github.com/hiyouga/LLaMA-Factory`, 2023. 7

Aspen K Hopkins, Alex Renda, and Michael Carbin. Can llms generate random numbers? evaluating llm sampling in controlled domains. In *ICML 2023 Workshop: Sampling and Optimization in Discrete Space*, 2023. 2

Edward J Hu, Moksh Jain, Eric Elmoznino, Younesse Kaddar, Guillaume Lajoie, Yoshua Bengio, and Nikolay Malkin. Amortizing intractable inference in large language models. *arXiv preprint arXiv:2310.04363*, 2023. 2

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. 6

Jing Jiang and ChengXiang Zhai. Instance weighting for domain adaptation in nlp. In *Proceedings of the 45th Annual Meeting of the Association Computational Linguistics*. ACL, 2007. 9

Denis Kocetkov, Raymond Li, Loubna Ben Allal, Jia Li, Chenghao Mou, Carlos Muñoz Ferrandis, Yacine Jernite, Margaret Mitchell, Sean Hughes, Thomas Wolf, et al. The stack: 3 tb of permissively licensed source code. *arXiv preprint arXiv:2211.15533*, 2022. 1, 4, 9

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213, 2022. 3

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023. 5

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857, 2022. 1, 9

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*, 2023. 1, 2, 9

Haoxiong Liu, Yifan Zhang, Yifan Luo, and Andrew Chi-Chih Yao. Augmenting math word problems via iterative question composing. *arXiv preprint arXiv:2401.09003*, 2024. 5

Miaofeng Liu, Yan Song, Hongbin Zou, and Tong Zhang. Reinforced training data selection for domain adaptation. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 1957–1968, 2019. 9

Sören Mindermann, Jan M Brauner, Muhammed T Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, et al. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning*, pp. 15630–15649. PMLR, 2022. 9

Robert C Moore and William Lewis. Intelligent selection of language model training data. In *Proceedings of the ACL 2010 conference short papers*, pp. 220–224, 2010. 9

OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023. 1

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022. 2

Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. Openwebmath: An open dataset of high-quality mathematical web text. *arXiv preprint arXiv:2310.06786*, 2023. 1, 2, 4, 5, 7, 9

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. *openai.com*, 2018. 1

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 1, 9

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023. 2

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: memory optimizations toward training trillion parameter models. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '20. IEEE Press, 2020. ISBN 9781728199986. 7

Laria Reynolds and Kyle McDonell. Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–7, 2021. 3

Sebastian Ruder and Barbara Plank. Learning to select data for transfer learning with bayesian optimization. *arXiv preprint arXiv:1707.05246*, 2017. 9

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*, 2022. 2, 3, 8

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022. 9

Zengzhi Wang, Rui Xia, and Pengfei Liu. Generative ai for math: Part i–mathpile: A billion-token-scale pretraining corpus for math. *arXiv preprint arXiv:2312.17120*, 2023. 1, 5

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022. 3, 9

Kai Wei, Rishabh Iyer, and Jeff Bilmes. Submodularity in data subset selection and active learning. In *International conference on machine learning*, pp. 1954–1963. PMLR, 2015. 9

Alexander Wettig, Aatmik Gupta, Saumya Malik, and Danqi Chen. Qurating: Selecting high-quality data for training language models. *arXiv preprint arXiv:2402.09739*, 2024. 6, 7, 9

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. *arXiv preprint arXiv:2305.10429*, 2023a. 9

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. Data selection for language models via importance resampling. *arXiv preprint arXiv:2302.03169*, 2023b. 6, 9

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *arXiv preprint arXiv:2305.10601*, 2023. 9

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*, 2023. 7

Yifan Zhang, Jingqin Yang, Yang Yuan, and Andrew Chi-Chih Yao. Cumulative reasoning with large language models. *arXiv preprint arXiv:2308.04371*, 2023a. 9

Yifan Zhang, Yang Yuan, and Andrew Chi-Chih Yao. Meta prompting for ai systems. *arXiv preprint arXiv:2311.11482*, 2023b. 3

A Appendix for Examples

A.1 Web Subset

Example: Commutative Property Of Addition
"Commutative Property Of Addition 2. If A is an n×m matrix and O is a m×k zero-matrix, then we have: AO = O Note that AO is the n×k zero-matrix. Matrix Matrix Multiplication 11:09. We have 1. To understand the properties of transpose matrix, we will take two matrices A and B which have equal order. The identity matrix is a square matrix that has 1's along the main diagonal and 0's for all other entries. In a triangular matrix, the determinant is equal to the product of the diagonal elements. This matrix is often written simply as I, and is special in that it acts like 1 in matrix multiplication. Is the Inverse Property of Matrix Addition similar to the Inverse Property of Addition? The identity matrices (which are the square matrices whose entries are zero outside of the main diagonal and 1 on the main diagonal) are identity elements of the matrix product. Learning Objectives. In fact, this tutorial uses the Inverse Property of Addition and shows how it can be expanded to include matrices! Keywords: matrix; matrices; inverse; additive; additive inverse; opposite; Background Tutorials. ..."
LM-Score (Q_1): 0.991, **LM-Score (Q_2):** 0.954, **LM-Score (Q_1, Q_2):** 0.946

Example: Comparing the magnitudes of expressions
"# Comparing the magnitudes of expressions of surds I recently tackled some questions on maths-challenge / maths-aptitude papers where the task was to order various expressions made up of surds (without a calculator, obviously). I found myself wondering whether I was relying too much on knowing the numerical value of some common surds, when a more robust method was available (and would work in more difficult cases). For example, one question asked which is the largest of: (a) $\sqrt{10}$ (b) $\sqrt{2} + \sqrt{3}$ (c) $5 - \sqrt{3}$ In this case, I relied on my knowledge that $\sqrt{10} \approx 3.16$ and $\sqrt{2} \approx 1.41$ and $\sqrt{3} \approx 1.73$ to find (a) $\approx$ 3.16, (b) $\approx$ 3.14 and (c) $\approx$ 3.27 so that the required answer is (c). But this seemed inelegant: I felt there might be some way to manipulate the surd expressions to make the ordering more explicit. I can't see what that might be, however (squaring all the expressions didn't really help). ..."
LM-Score (Q_1): 0.991, **LM-Score (Q_2):** 0.946, **LM-Score (Q_1, Q_2):** 0.937

Example: In Calculus, function derivatives
"# In Calculus, how can a function have several different, yet equal, derivatives? I've been pondering this question all night as I work through some problems, and after a very thorough search, I haven't found anything completely related to my question. I guess i'm also curious how some derivatives are simplified as well, because in some cases I just can't see the breakdown. Here is an example: $f(x) = \dfrac{x^2 - 6x + 12}{x - 4}$ is the function I was differentiating. Here is what I got: $f'(x) = \dfrac{x^2 - 8x + 12}{(x-4)^2}$ which checks using desmos graphing utility. Now, when I checked my textbook(and Symbolab) they got: $f'(x) = 1 - \dfrac{4}{(x-4)^2}$ which also checks on desmos. To me, these derivatives look nothing alike, so how can they both be the equal to the derivative of the original function? Both methods used the quotient rule, yet yield very different results. Is one of these "better" than the other? I know that it is easier to find critical numbers with a more simplified derivative, but IMO the derivative I found seems easier to set equal to zero than the derivative found in my book.I also wasn't able to figure out how the second derivative was simplified, so I stuck with mine. I'm obviously new to Calculus and i'm trying to understand the nuances of derivatives. ..."
LM-Score (Q_1): 0.985, **LM-Score (Q_2):** 0.950, **LM-Score (Q_1, Q_2):** 0.936

Example: Math help on cubics

"# Math Help - working backwards - cubics 1. ## working backwards - cubics Write an equation that has the following roots: 2, -1, 5 Answer key: $x^3 - 6x^2 + 3x + 10 = 0$ For quadratic equations, I use the sum and product of roots, this is a cubic equation, how do I solve this? Thanks. 2. Originally Posted by shenton Write an equation that has the following roots: 2, -1, 5 Answer key: $x^3 - 6x^2 + 3x + 10 = 0$ For quadratic equations, I use the sum and product of roots, this is a cubic equation, how do I solve this? Thanks. $(x-2)(x+1)(x-5)$ 3. Thanks! That turns out to be not as difficult as imagined. I thought I needed to use sum and products of roots to write the equation, it does makes me wonder a bit why or when I need to use sum and products of roots. 4. Write an equation that has the following roots: 2, -1, 5 Is there any other way to solve this other than the (x-2)(x+1)(x-5) method? If we have these roots: 1, $1+\sqrt{2}$, $1-\sqrt{2}$ the (x - 1) $(x-1-\sqrt{2})$ $(x-1+\sqrt{2})$ method seems a bit lenghty. When we expand (x - 1) $(x-1-\sqrt{2})$ $(x-1+\sqrt{2})$ the first 2 factors, it becomes: $(x^2 - x - x\sqrt{2} - x + 1 + \sqrt{2})$ $(x-1+\sqrt{2})$ collect like terms: $(x^2 - 2x - x\sqrt{2} + 1 + \sqrt{2})$ $(x-1+\sqrt{2})$ To further expand this will be lenghty, my gut feel is that mathematicians do not want to do this - it is time consuming and prone to error. There must be a way to write an equation other than the above method. Is there a method to write an equation with 3 given roots (other than the above method)? ..."

LM-Score (Q_1): 0.991, **LM-Score (Q_2):** 0.943, **LM-Score (Q_1, Q_2):** 0.935

Example: Work and time

"# Work and time, when work is split into parts I'm stuck on a particular type of work and time problems. For example, 1) A,B,C can complete a work separately in 24,36 and 48 days. They started working together but C left after 4 days of start and A left 3 days before completion of the work. In how many days will the work be completed? A simpler version of the same type of problem is as follows: 2) A can do a piece of work in 14 days while B can do it in 21 days. They begin working together but 3 days before the completion of the work, A leaves off. The total number of days to complete the work is? My attempt at problem 2: A's 1 day work=1/14 and B's 1 day work= 1/21 Assume that it takes 'd' days to complete the entire work when both A and B are working together. Then, (1/14 + 1/21)*d= 1 -¿ d=42/5 days. But it is stated that 3 days before the completion of the work, A left. Therefore, work done by both in (d-3) days is: (1/14 + 1/21)*(42/5 - 3)= 9/14 Remaining work= 1- 9/14 = 5/14 which is to be done by B alone. Hence the time taken by B to do (5/14) of the work is: (5/14)*21 = 7.5 days. Total time taken to complete the work = (d-3) + 7.5 = 12.9 days. However, this answer does not concur with the one that is provided. My Understanding of problem 1: Problem 1 is an extended version of problem 2. But since i think i'm doing problem 2 wrong, following the same method on problem 1 will also result in a wrong answer. Where did i go wrong? ..."

LM-Score (Q_1): 0.991, **LM-Score (Q_2):** 0.941, **LM-Score (Q_1, Q_2):** 0.932

Example: Inequality Involving Sums

"Inequality involving sums with binomial coefficient I am trying to show upper- and lower-bounds on $\frac{1}{2^n}\sum_{i=0}^{n}\binom{n}{i}\min(i, n-i)$ (where $n \geq 1$) in order to show that it basically grows as $\Theta(n)$. The upper-bound is easy to get since $\min(i, n-i) \leq i$ for $i \in \{0, \dots n\}$ so that $\frac{1}{2^n}\sum_{i=0}^{n}\binom{n}{i}\min(i, n-i) \leq \frac{1}{2^n}\sum_{i=0}^{n}\binom{n}{i}i = \frac{n}{2}$. Thanks to Desmos, I managed to find a lower bound, but I am struggling to actually prove it. Indeed, I can see that the function $f(n) = \frac{n-1}{3}$ does provide a lower-bound. One can in fact rewrite $\frac{n-1}{3} = \frac{1}{2^n}\sum_{i=0}^{n}\binom{n}{i}\frac{2i-1}{3}$. I was thus hoping to show that for each term we have $\frac{2i-1}{3} \leq \min(i, n-i)$, but this is only true if $i \leq \frac{3n+1}{5}$ and not generally for $i \leq n$. I imagine there is a clever trick to use at some point but for some reason, I am stuck here. Any help would be appreciated, thank you! EDIT: Thank you everyone for all the great and diverse answers! I flagged River Li's answer as the "accepted" one because of its simplicity due to the use of Cauchy-Schwartz inequality, which does not require a further use of Stirling's approximation. ..."

LM-Score (Q_1): 0.988, **LM-Score (Q_2):** 0.941, **LM-Score (Q_1, Q_2):** 0.931

Example: Finding the minimum number

"# Finding the minimum number of students There are p committees in a class (where $p \geq 5$), each consisting of q members (where $q \geq 6$).No two committees are allowed to have more than 1 student in common. What is the minimum and maximum number of students possible? It is easy to see that the maximum number of student is pq,however I am not sure how to find the minimum number of students.Any ideas? 1) $\quad pq - \binom{q}{2}$ 2) $\quad pq - \binom{p}{2}$ 3) $\quad (p-1)(q-1)$ - Something is missing. Is every student supposed to be on a committee? – JavaMan Aug 31 '11 at 16:24 @DJC:Not mentioned in the question,I guess we may have to consider that to get a solution. – Quixotic Aug 31 '11 at 16:28 @DJC: For the minimum number of students this does not matter. – TMM Aug 31 '11 at 16:30 @Thijs Laarhoven:Yes you are right but as the problem also asked for maximum number I have considered it in my solution. – Quixotic Aug 31 '11 at 16:31 @Thijs, FoolForMath, I guess my question is, should the minimum answer be in terms of p and q? – JavaMan Aug 31 '11 at 16:31 For $1 \leq i \leq p$, let C_i be the set of students on the ith committee. Then by inclusion-exclusion, or more accurately Boole's inequalities, we have

$$\sum_i |C_i| - \sum_{i<j} |C_i C_j| \leq |C_1 \cup C_2 \cup \cdots \cup C_p| \leq \sum_i |C_i|.$$

From the constraints of the problem, this means

$$pq - \binom{p}{2} \leq \# \text{ students} \leq pq.$$

- What is j here?and I can't relate this with your answer. j is also a generic index that runs from 1 to p. The inequalities are also known as Bonferroni inequalities (planetmath.org/encyclopedia/BonferroniInequalities.html), and can apply to cardinalities instead of probabilities. – Byron Schmuland Sep 1 '11 at 14:10 I think the following theorem might be relevant: Theorem. Let $\mathcal{F}$ be a family of subsets of $\{1, \ldots, n\}$ with the property that $|A \cap B| = 1$ for all $A, B \in \mathcal{F}$. Then $|\mathcal{F}| \leq n$. Also this theorem could be relevant as well. - For the case in which $p \leq q+1$ an arrangement that yields the minimum number of students can be described as follows. Let $P = \{\langle m, n \rangle : 1 \leq m \leq p, 1 \leq n \leq q+1\}$, and let $S = \{\langle m, n \rangle \in P : m < n\}$. If P is thought of as a $p \times (q+1)$ grid, ..."

LM-Score (Q_1): 0.985, **LM-Score (Q_2):** 0.863, **LM-Score (Q_1, Q_2):** 0.850

Example: Applied Linear Algebra

"Let $w_1 = (0, 1, 1)$. Expand $\{w_1\}$ to a basis of R^3. I am reading the book, Applied Linear Algebra and Matrix Analysis. When I was doing the exercise of Section3.5 Exercise 7, I was puzzled at some of it. Here is the problem description: Let $w_1 = (0, 1, 1)$. Expand $\{w_1\}$ to a basis of R^3. I don't understand its description well. I think it wants to get a span set like $\{(0, 1, 1), (1, 0, 0), (0, 0, 1)\}$ which is a basis of R^3. And I check the reference answer, which is as followings: $(0, 1, 1)$, $(1, 0, 0)$, $(0, 1, 0)$ is one choice among many. I think what I have done is what question wants. So can anyone tell me am I right or wrong? Thanks sincerely. • I think you are right Apr 16, 2019 at 6:02 There is a kind of 'procedure' for dealing with questions of this kind, namely to consider the spanning set $\{w_1, e_1, e_2, e_3\}$. Consider each vector from left to right. If one of these vectors is in the span of the previous one/s, then throw it out. If not, keep it. So in this case, we start by keeping w_1. Moving to the next vector, e_1 is not in the span of w_1, so we keep it as well. Moving to the next, e_2 is not in the span of the previous two vectors so we keep it as well. Now, considering the vector e_3 we see that it is in fact in the span of the previous three vectors, since $e_3 = w_1 - e_2$. So we throw out the vector e_3 and end up with the basis $\{w_1, e_1, e_2\}$. This explains the solution in the reference answer. Your solution is also correct, however. $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$ has independent rows. Hence you have found 3 independent vectors in $\mathbb{R}^3$, that is it spans $\mathbb{R}^3$ and it forms a basis. You are correct. $(0, 1, 1), (1, 0, 0), (0, 0, 1)$ is a basis of $\mathbb{R}^3$. Any element (a, b, c) in $\mathbb{R}^3$ can be expressed as $a(1, 0, 0) + b(0, 1, 1) + (c - b)(0, 0, 1)$. If your basis is w_1, w_2, w_3, the textbook's choice is $w_1, w_2, w_1 - w_3$ ..."

LM-Score (Q_1): 0.964, **LM-Score (Q_2):** 0.882, **LM-Score (Q_1, Q_2):** 0.850

Example: Vector equations

"# Vector equations, possible to solve for x? #### Jonsson Hello there, In scalar algebra, I find solving for variables a useful tool. Say ohms law, I want to find R so:

$$U = RI \iff R = \frac{U}{I}$$

Can I do something analogous in vector equations? I.e. May I solve for $\vec{\omega}$ in equations using cross or dot products?

$$\vec{v} = \vec{\omega} \times \vec{r} \iff \vec{\omega} = ?$$

or:

$$\vec{\alpha} \cdot \vec{\beta} = \gamma \iff \vec{\beta} = ?$$

It would be fantastic if I could solve for vectors in some way. Hope you are able to help. Kind regards, Marius #### maajdl Gold Member Solving v=wxr makes sense, since this can be seen as solving 3 equations with 3 unknowns (each components). You can find the solution easily by "multiplying" both sides by r: rxv = rx(wxr) = w (r.r) - r (w.r). ..."

LM-Score (Q_1): 0.950, **LM-Score (Q_2):** 0.842, **LM-Score (Q_1, Q_2):** 0.800

Example: Linear programming

"# If then Constraint 2 Hello all: I want to implement the following constraint in my linear programing model: If A=B then C=1 Else C=0 I have been looking around and there are similar problems but nobody has been helpful to address the 'non equal to' condition. Thank you in advance. asked 27 Sep '14, 17:45 Chicago 33 5 accept rate: 0% 3 As I understand the question, you want c to be binary, and $c = 1$ if and only if $A = B$. I will make a couple of assumptions: There is a (large) positive M such that $|A - B| \leq M$ for every feasible (A, B). There is a (small) positive ϵ such that whenever $A \neq B$, we can assume there is a solution satisfying $|A - B| \geq \epsilon$. Here's the formulation:

$$\begin{aligned} A &\leq B + My - \epsilon z \\ B &\leq A + Mz - \epsilon y \\ c + y + z &= 1 \\ c, y, z &\in \{0, 1\} \end{aligned}$$

Now, if $c = 1$, then $y = z = 0$. In this case, the constraints reduce to $A \leq B$ and $B \leq A$, so $A = B$. Otherwise, $c = 0$. Then $y + z = 1$. There are two cases. ..."

LM-Score (Q_1): 0.950, **LM-Score (Q_2):** 0.842, **LM-Score (Q_1, Q_2):** 0.800

Example: Distance formula

"The distance formula is a formula that is used to find the distance between two points. These points can be in any dimension. The x-z plane is vertical and shaded pink ... If observation i in X or observation j in Y contains NaN values, the function pdist2 returns NaN for the pairwise distance between i and j.Therefore, D1(1,1), D1(1,2), and D1(1,3) are NaN values.. Contents. Print the the distance between two points on the surface of earth: —– Input the latitude of coordinate 1: 25 Input the longitude of coordinate 1: 35 Input the latitude of coordinate 2: 35.5 Input the longitude of coordinate 2: 25.5 The distance between those points is: 1480.08 Flowchart: C++ Code Editor: Contribute your code and comments through Disqus. Interactive Distance Formula applet. Distance Formula Calculator. Find the square root of that sum: $\sqrt{90} = 9.49$. In a 3 dimensional plane, the distance between points (X 1, Y 1, Z 1) and (X 2, Y 2, Z 2) are given. The distance between two points on the three dimensions of the xyz-plane can be calculated using the distance formula The distance formula is derived from the Pythagorean theorem. and: Line passing through two points. Parameters first Iterator pointing to the initial element. Distance between 2 points in 3D space calculator uses Distance between 2 points=$\sqrt{(x2 - x1)^2 + (y2 - y1)^2 + (z2 - z1)^2}$ to calculate the Distance between 2 points, ..."

LM-Score (Q_1): 0.950, **LM-Score (Q_2):** 0.737, **LM-Score (Q_1, Q_2):** 0.700

Example: Estimate from below of the sine

"# Estimate from below of the sine (and from above of cosine) I'm trying to do the following exercise with no success. I'm asked to prove that $\sin(x) \geq x - \frac{x^3}{2}$, $\quad \forall x \in [0, 1]$ By using Taylor's expansion, it's basically immediate that one has the better estimate $\sin(x) \geq x - \frac{x^3}{6}$, $\quad \forall x \in [0, 1]$ as the tail converges absolutely, and one can check that the difference of consecutive terms is positive. I suppose then, there is a more elementary way to get the first one. Question is: how? Relatedly, the same exercise asks me to prove that $\cos(x) \leq \frac{1}{\sqrt{1+x^2}}$, $\quad \forall x \in [0, 1]$ which again I can prove by using differentiation techniques. But these haven't been explained at that point of the text, so I wonder how to do it "elementary". I showed by comparison of areas that for first quadrant angles $\sin\theta\cos\theta \leq \theta \leq \tan\theta$ If one multiplies the left of these inequalities by 2 it becomes $\sin 2\theta < 2\theta$ so we arrive at $\sin\theta \leq \theta \leq \tan\theta$ Rearrange the right of these inequalities to $\frac{\sin\theta}{\theta} \geq \cos\theta$ or $1 - \frac{\sin\theta}{\theta} \leq 1 - \cos\theta = 2\sin^2\frac{\theta}{2} \leq 2\left(\frac{\theta}{2}\right)^2 = \frac{\theta^2}{2}$ Where we have used the left of the above inequalities above. This rearranges to $\sin\theta \geq \theta - \frac{\theta^3}{2}$ for first quadrant angles. ..."

LM-Score (Q_1): 0.950, **LM-Score (Q_2):** 0.737, **LM-Score (Q_1, Q_2):** 0.700

Example: Force on side of pool from water

"Force on side of pool from water Given a pool with dimensions $\ell \times w \times h$, I am trying to derive an equation that will yield the force by the water on the sides of the pool, namely $\ell \times h \quad$ or $\quad w \times h$. For the side of the pool with dimensions $\ell \times h$, I started by using the familiar equation for pressure $F = PA$. Plugging in the expression for hydrostatic pressure for P gives $F = \rho ghA = \rho gh(\ell \times h) = \boxed{\rho g\ell h^2}$. Is my reasoning, and corresponding solution correct? Hydrostatic pressure changes with height. You have just multiplied by area, which means that you have assumed it to be constant. Instead, you should integrate over the area. You'll get an extra 1/2 term for the force. – Goobs Sep 15 '15 at 4:21 As @Goobs says, the pressure force is 0 at the top of the water line and increases to $\rho\ g\ y\ dA$ on a surface of area dA at depth y. Since this pressure increases linearly from 0 to $\rho\ g\ y$ the average force on the wall is the average of the start and end: so, it is half of this value, and the total pressure is $\frac{1}{2}\rho gh(h\ell)$. Would this be correct? $\int dF = \int_0^H \rho gA\ dh = \rho g\ell \int_0^H h\ dh = \boxed{\frac{1}{2}\rho gH^2}$ – rgarci0959 Sep 15 '15 at 4:51 Yes. For bonus points you would write it as $\int dA\ \rho\ g\ h$ to start with, as that's one of those forces that you "know" is correct ..."

LM-Score (Q_1): 0.987, **LM-Score (Q_2):** 0.662, **LM-Score (Q_1, Q_2):** 0.653

Example:

"# In mathematics the monomial basis of a polynomial ring is its basis (as vector space or free module over the field or ring of coefficients) that consists in the set of all monomials. The monomials form a basis because every polynomial may be uniquely written as a finite linear combination of monomials (this is an immediate consequence of the definition of a polynomial). One indeterminate The polynomial ring K[x] of the univariate polynomial over a field K is a K-vector space, which has $1, x, x^2, x^3, \ldots$ as an (infinite) basis. More generally, if K is a ring, K[x] is a free module, which has the same basis. The polynomials of degree at most d form also a vector space (or a free module in the case of a ring of coefficients), which has $1, x, x^2, \ldots$ as a basis The canonical form of a polynomial is its expression on this basis: $a_0 + a_1x + a_2x^2 + \ldots + a_dx^d$, or, using the shorter sigma notation: $\sum_{i=0}^d a_ix^i$. The monomial basis in naturally totally ordered, either by increasing degrees $1 < x < x^2 < \cdots$, or by decreasing degrees $1 > x > x^2 > \cdots$. Several indeterminates In the case of several indeterminates $x_1, \ldots, x_n$, a monomial is a product $x_1^{d_1}x_2^{d_2}\cdots x_n^{d_n}$, where the d_i are non-negative integers. Note that, as $x_i^0 = 1$, an exponent equal to zero means that the corresponding indeterminate does not appear in the monomial; in particular $1 = x_1^0x_2^0\cdots x_n^0$ is a monomial. ..."

LM-Score (Q_1): 0.987, **LM-Score (Q_2):** 0.662, **LM-Score (Q_1, Q_2):** 0.653

"

Define a function called `isOdd` that takes an argument, $n \in \mathbb{N}$, and returns a proposition that asserts that n is odd. The function will thus be a predicate on values of type $\mathbb{N}$. Hint: a number is odd if it's one more than an even number.

$$\text{def isOdd}(n : \mathbb{N}) : \text{Prop} := \exists m : \text{nat}, 2 \cdot m + 1 = n$$

To test your predicate, use "example" to write and prove `isOdd(15)`.

```
example : isOdd 15 :=
begin
unfold isOdd,
apply exists.intro 7,
apply rfl,
end
```

Define `isSmall` : $\mathbb{N} \rightarrow$ Prop, to be a predicate that is true exactly when the argument, n, is such that $n = 0 \vee n = 1 \vee n = 2 \vee n = 3 \vee n = 4 \vee n = 5$. (Don't try to rewrite this proposition as an inequality; just use it as is.)

$$\text{def isSmall}(n : \mathbb{N}) : \text{Prop} := n = 0 \vee n = 1 \vee n = 2 \vee n = 3 \vee n = 4 \vee n = 5$$

"

...

[LM-Score (Q_1, Q_2): 0.963]

"

Define the universes and variables for the context of our category and functor:

$$\textbf{universes } v\ u$$

$$\textbf{variables } \{J : \text{Type } v\} \text{ [small\_category } J] \{C : \text{Type } u\} \text{ [category.}\{v\}\ C] (F : J \rightarrow C)$$

Enter noncomputable theory mode and define the initial object's colimit cocone:

```
def is_initial.colimit_cocone {j : J} (hj : is_initial j)
  [has_colimit F] [\forall (a b : J) (f : a \rightarrow b),
  is_iso (F.map f)] :

  cocone F :=
{ X := F.obj j,
  \iota :=
  { app := $\lambda$ i, inv (F.map $ hj.to _),
    naturality' := begin
      intros a b f,
      dsimp,
      simp only [is_iso.eq_inv_comp, is_iso.comp_inv_eq,
      category.comp_id],
      simp_rw ← F.map_comp,
      congr' 1,
      apply hj.hom_ext,
    end } }
```

"

...

[LM-Score (Q_1, Q_2): 0.439]

Figure 7: Examples contain Lean4 code. It is difficult for human beings without math expertise to judge the educational value of these examples for language models on learning mathematics.

A.2 Code Subset

Example: Lagrange's Interpolation Method

```
X = [0, 20, 40, 60, 80, 100]
Y = [26.0, 48.6, 61.6, 71.2, 74.8, 75.2]
n = len(X)-1
# Degree of polynomial = number of points - 1
print("X =", X)
print("Y =", Y, end='\n\n')
xp = float(input("Find Y for X = "))
# For degree of polynomial 3, number of points n+1 = 4:
# L[1] = (x-x2)/(x1-x2) * (x-x3)/(x1-x3) * (x-x4)/(x1-x4)
# L[2] = (x-x1)/(x2-x1) * (x-x3)/(x2-x3) * (x-x4)/(x2-x4)
# L[3] = (x-x1)/(x3-x1) * (x-x2)/(x3-x2) * (x-x4)/(x3-x4)
# L[4] = (x-x1)/(x4-x1) * (x-x2)/(x4-x2) * (x-x3)/(x4-x3)
# L[i] *= (x-xj)/(xi-xj) where i, j = 1 to n+1 and j != i
# y += Y[i]*L[i] where i = 1 to n+1
# List index 0 to n
# ~~~~~~~~~~~~~~~~~~~~~~~~ Method 1: Using for loop ~~~~~~~~~~~~~~~~~~~~~~~~
yp = 0
# Initial summation value
for i in range(n+1):
    L = 1
    # Initial product value
    for j in range(n+1):
        if j == i:
            continue
        # j == i gives ZeroDivisionError
        L *= (xp - X[j]) / (X[i] - X[j]) yp += Y[i]*L
# ~~~~~~~~~~~~~~~~~~~ Method 2: Using numpy array, prod ~~~~~~~~~~~~~~~~~~~~
from numpy import array, prod
X = array(X, float)
Y = array(Y, float)
yp = 0
for Xi, Yi in zip(X, Y):
    yp += Yi * prod((xp - X[X != Xi]) / (Xi - X[X != Xi]))
```

LM-Score (Q_1): 0.977, **LM-Score (Q_2):** 0.959, **LM-Score (Q_1, Q_2):** 0.937

Example: Scientific Computing Theory

```
# Question 01, Lab 04
# AB Satyaprakash - 180123062
# imports ----------------------------------------------------------------------------
from sympy.abc import x
from sympy import cos, exp, pi, evalf, simplify
# functions --------------------------------------------------------------------------
def midpointRule(f, a, b):
    return ((b-a)*f.subs(x, (b-a)/2)).evalf()

def trapezoidalRule(f, a, b):
    return (((b-a)/2)*(f.subs(x, a)+f.subs(x, b))).evalf()

def simpsonRule(f, a, b):
    return (((b-a)/6)*(f.subs(x, a)+4*f.subs(x, (a+b)/2)+f.subs(x, b))).evalf()

# program body
# part (a) I = integrate cosx/(1+cos^2x) from 0 to pi/2 -- exact value = 0.623225
f = cos(x)/(1 + cos(x)**2)
a, b = 0, pi/2
print('To integrate {} from {} to {}'.format(simplify(f), a, b))
print('Evaluated value of integral using Midpoint rule is', midpointRule(f, a, b))
print('Evaluated value of integral using Trapezoidal rule is', trapezoidalRule(f, a, b))
print('Evaluated value of integral using Simpson rule is', simpsonRule(f, a, b))
print('Exact value = 0.623225\n')

# part (b) I = integrate 1/(5+4cosx) from 0 to pi -- exact value = 1.047198
f = 1/(5 + 4*cos(x))
a, b = 0, pi
print('To integrate {} from {} to {}'.format(simplify(f), a, b))
print('Evaluated value of integral using Midpoint rule is', midpointRule(f, a, b))
print('Evaluated value of integral using Trapezoidal rule is', trapezoidalRule(f, a, b))
print('Evaluated value of integral using Simpson rule is', simpsonRule(f, a, b))
print('Exact value = 1.047198\n')

# part (c) I = integrate exp(-x^2) from 0 to 1 -- exact value = 0.746824
f = exp(-x**2)
a, b = 0, 1
```

LM-Score (Q_1): 0.982, **LM-Score (Q_2):** 0.946, **LM-Score (Q_1, Q_2):** 0.929

Example: Fourth Order Runge-Kutta (RK4) Method

```
from numpy import exp, linspace, empty
f = lambda x: exp(x-2) - 3 # Analytical Solution
dy = lambda x, y: y+3 # Equation to be solved, y' = y+3
x = 2 # Lower limit, [2
xn = 4 # Upper limit, 4]
y = -2 # Initial condition, y(2) = -2
h = 0.1 # Width of each division, step size
n = int((xn-x)/h) # Number of divisions of the domain
# Plot Arrays
xp = linspace(x, xn, n+1)
# Divides from x to xn into n+1 points
yp = empty(n+1, float)
yp[0] = y
print('x \t\ty(RK4) \t\ty(Analytical)')
# Header of Output
print('%f \t% f \t% f' % (x, y, f(x)))
# Initial x and y
for i in range(1, n+1):
    K1 = h * dy(x,y)
    K2 = h * dy(x + h/2, y + K1/2)
    K3 = h * dy(x + h/2, y + K2/2)
    K4 = h * dy(x + h, y + K3)
    y += 1/6*(K1 + 2*K2 + 2*K3 + K4) # y(x+h) = y(x) + 1/6(K1+2K2+2K3+K4)
    yp[i] = y
    x += h # x for next step,
    x = x + h
    print('%f \t% f \t% f' % (x, y, f(x)))
    # ~~~~~~~~~~~~~~~~~~~~~~~~~ Plotting the function ~~~~~~~~~~~~~~~~~~~~~~~~~~
    import matplotlib.pyplot as plt # pyplot.
    plt.plot(xp, yp, 'ro', xp, f(xp)) # Default plot is continuous blue line
    plt.xlabel('x')
    plt.ylabel('y')
    plt.legend(['RK4', 'Analytical'])
    plt.show()
```

LM-Score (Q_1): 0.982, **LM-Score (Q_2):** 0.945, **LM-Score (Q_1, Q_2):** 0.928

Example: Real roots of the quadratic equation

```
from math import sqrt
from numpy.testing import assert_equal, assert_allclose
def real_quadratic_roots(a, b, c):
"""
Find the real roots of the quadratic equation a x^2 + b x + c = 0, if they exist.
Parameters ----------
a : float Coefficient of x^2
b : float Coefficient of x^1
c : float Coefficient of x^0
Returns -------
roots : tuple or float or None The root(s) (two if a genuine quadratic, one if linear, None
  otherwise)
Raises ------
NotImplementedError If the equation has trivial a and b coefficients, so isn't solvable.
"""
    discriminant = b**2 - 4.0*a*c
    if discriminant < 0.0:
        return None
    if a == 0:
        if b == 0:
            raise NotImplementedError("Cannot solve quadratic with both a" " and b
  coefficients equal to 0.")
        else: return -c / b

    x_plus = (-b + sqrt(discriminant)) / (2.0*a)
    x_minus = (-b - sqrt(discriminant)) / (2.0*a)
    return x_plus, x_minus

def test_no_roots():
"""
Test that the roots of x^2 + 1 = 0 are not real.
"""
roots = None
assert_equal(real_quadratic_roots(1, 0, 1), roots, err_msg="Testing x^2+1=0; no real roots.")
```

LM-Score (Q_1): 0.977, **LM-Score (Q_2):** 0.950, **LM-Score (Q_1, Q_2):** 0.928

A.3 ARXIV SUBSET

Example: "Convergence directions of the randomized Gauss–Seidel method and its extension"
"... Linear least squares problem is a ubiquitous problem arising frequently in data analysis and scientific computing. Specifically, given a data matrix $A \in R^{m \times n}$ and a data vector $b \in R^m$, a linear least squares problem can be written as follows

$$\min_{x \in R^n} \|b - Ax\|_2^2. \tag{3}$$

In the literature, several direct methods have been proposed for solving its normal equations $A^T Ax = A^T b$ through either the QR factorization or the singular value decomposition (SVD) of $A^T A$ (bjorck1996numerical, Higham2002), which can be prohibitive when the matrix is large–scale. Hence, iterative methods are considered for solving large linear least squares problem, such as the famous Gauss–Seidel method (Saad2003). In (Leventhal2010), Leventhal and Lewis proved that the randomized Gauss–Seidel (RGS) method, also known as the randomized coordinate descent method, converges to the solution at a linear rate in expectation. This method works on the columns of the matrix A at random with probability proportional to their norms. Later, Ma, Needell and Ramdas (Ma2015) provided a unified theory of the RGS method and the randomized Kaczmarz (RK) method (Strohmer2009), where the latter method works on the rows of A, and showed that the RGS method converges to the minimum Euclidean norm least squares solution $x_\star$ of (3) only when the matrix A is of full column rank. To further develop the RGS method for more general matrix, inspired by the randomized extended Kaczmarz (REK) method (Completion2013), Ma et al. (Ma2015) presented a variant of the RGS mehtod, ..."
LM-Score (Q_1): 0.991, **LM-Score (Q_2):** 0.818, **LM-Score (Q_1, Q_2):** 0.810

Example: "A fixed point theorem for the infinite-dimensional simplex"
"... In finite dimensions, one of the simplest methods for proving the Brouwer fixed point theorem is via a combinatorial result known as Sperner's lemma (Sper28), which is a statement about labelled triangulations of a simplex in $\mathbb{R}^n$. In this paper, we use Sperner's lemma to prove a fixed point theorem on an infinite-dimensional simplex in $\mathbb{R}^\infty$. We also show that this theorem implies the infinite-dimensional case of Schauder's fixed point theorem on normed spaces. Since $\mathbb{R}^\infty$ is locally convex, our theorem is a consequence of Tychonoff's fixed point theorem (Smar74). However, some notable advantages of our approach are: (1) the constructive nature of Sperner's lemma provides a method for producing approximate fixed points for functions on the infinite-dimensional simplex, (2) the proof is based on elementary methods in topology and analysis, and (3) our proof provides another route to Schauder's theorem. Fixed point theorems and their constructive proofs have found many important applications, ranging from proofs of the Inverse Function Theorem (Lang97), to proofs of the existence of equilibria in economics (Todd76, Yang99), to the existence of solutions of differential equations (Brow93, Smar74).
Working in $\mathbb{R}^\infty$ Let $\mathbb{R}^\infty$ and $I^\infty = \prod[0,1]$ be the product of countably many copies of $\mathbb{R}$, and $I = [0,1]$, respectively. We equip $\mathbb{R}^\infty$ with the standard product topology, which is metrizable (BePe75) by the complete metric

$$\bar{d}(x,y) = \sum_{i=1}^{\infty} \frac{|x_i - y_i|}{2^i(1 + |x_i - y_i|)}.$$

In $\mathbb{R}^n$, a k-dimensional simplex, or *k-simplex*, σ^k is the convex hull of $k+1$ affinely independent points. The *standard n-simplex* in $\mathbb{R}^{n+1}$, denoted Δ^n, is the convex hull of the $n+1$ standard basis vectors of $\mathbb{R}^n$. The natural extension of this definition to $\mathbb{R}^\infty$ is to consider Δ^∞, the convex hull of the standard basis vectors $\{e_i\}$ in $\mathbb{R}^\infty$, where $(e_i)_j = \delta_{ij}$, the Kronecker delta function. ..."
LM-Score (Q_1): 0.974, **LM-Score (Q_2):** 0.831, **LM-Score (Q_1, Q_2):** 0.810

Example: On connectedness of power graphs of finite groups

"Study of graphs associated to algebraic structures has a long history. There are various graphs constructed from groups and semigroups, e.g., Cayley graphs (cayley1878desiderata, budden1985cayley), intersection graphs (MR3323326, zelinka1975intersection), and commuting graphs (bates2003commuting). Kelarev and Quinn (kelarev2000combinatorial, kelarevDirectedSemigr) introduced the notion of *directed power graph* of a semigroup S as the directed graph $\overrightarrow{\mathcal{G}}(S)$ with vertex set S and there is an arc from a vertex u to another vertex v if $v = u^{\alpha}$ for some natural number $\alpha \in \mathbb{N}$. Followed by this, Chakrabarty et al. (GhoshSensemigroups) defined (*undirected*) *power graph* $\mathcal{G}(S)$ of a semigroup S as the (undirected) graph with vertex set S and distinct vertices u and v are adjacent if $v = u^{\alpha}$ for some $\alpha \in \mathbb{N}$ or $u = v^{\beta}$ for some $\beta \in \mathbb{N}$. Several authors studied power graphs and proved many interesting results. Some of them even exhibited the properties of groups from the viewpoint of power graphs. Chakrabarty (GhoshSensemigroups) et al. proved that the power graph of a finite group is always connected. They also showed that the power graph of a finite group G is complete if and only if G is a cyclic group of order 1 or p^k, for some prime p and $k \in \mathbb{N}$. Cameron and Ghosh observed isomorphism properties of groups based on power graphs. In (Ghosh), they showed that two finite abelian groups with isomorphic power graphs are isomorphic. Further, if two finite groups have isomorphic directed power graphs, then they have same numbers of elements of each order. Cameron (Cameron) proved that if two finite groups have isomorphic power graphs, then their directed power graphs are also isomorphic. It was shown by Curtin and Pourgholi that among all finite groups of a given order, the cyclic group of that order has the maximum number of edges and has the largest clique in its power graph (curtin2014edge,curtin2016euler). It was observed in (doostabadi2013some) and (MR3266285) that the power graph of a group is perfect. Perfect graphs are those with the same chromatic number and clique number for each of their induced subgraphs. Shitov (MR3612206) showed that for any group G, the chromatic number of $\mathcal{G}(G)$ is at most countable. ..."

LM-Score (Q_1): 0.985, **LM-Score (Q_2):** 0.803, **LM-Score (Q_1, Q_2):** 0.790

Example: Communication-optimal parallel and sequential QR and LU factorizations

"In this section, we review known lower bounds on communication bandwidth for parallel and sequential $\Theta(n^3)$ matrix-matrix multiplication of matrices stored in 2-D layouts, extend some of them to the rectangular case, and then extend them to LU and QR, showing that our sequential and parallel CAQR algorithms have optimal communication complexity with respect to both bandwidth (in a Big-Oh sense, and sometimes modulo polylogarithmic factors). We will also use the simple fact that if B is a lower bound on the number of words that must be communicated to implement an algorithm, and if W is the size of the local memory (in the parallel case) or fast memory (in the sequential case), so that W is the largest possible size of a message, then B/W is a lower bound on the latency, i.e. the number of messages needed to move B words into or out of the memory. We use this to derive lower bounds on latency, which are also attained by our algorithms (again in a Big-Oh sense, and sometimes modulo polylogarithmic factors). We begin in section MMlowerbounds by reviewing known communication complexity bounds for $\Theta(n^3)$ matrix multiplication, due first to Hong and Kung (hong1981io) in the sequential case, and later proved more simply and extended to the parallel case by Irony, Toledo and Tiskin (irony2004communication). It is easy to extend lower bounds for matrix multiplication to lower bounds for LU decomposition via the following reduction of matrix multiplication to LU:

$$\begin{pmatrix} I & 0 & -B \\ A & I & 0 \\ 0 & 0 & I \end{pmatrix} = \begin{pmatrix} I & & \\ A & I & \\ 0 & 0 & I \end{pmatrix} \begin{pmatrix} I & 0 & -B \\ & I & A \cdot B \\ & & I \end{pmatrix}. \tag{4}$$

See (grigori2008calu) for an implementation of parallel LU that attains these bounds. See (toledo1997locality) for an implementation of sequential LU and a proof that it attains the bandwidth lower bound (whether the latency lower bound is attained is an open problem). It is reasonable to expect that lower bounds for matrix multiplication will also apply (at least in a Big-Oh sense) to other one-sided factorizations, such as QR. ..."

LM-Score (Q_1): 0.970, **LM-Score (Q_2):** 0.815, **LM-Score (Q_1, Q_2):** 0.790

B More on Experiments

B.1 Prompts

"
<system>
You are ChatGPT, the most capable large language model equipped with extensive expertise in mathematics and coding, particularly skilled in complex reasoning and problem-solving. In the following interaction, I will provide you with a text excerpt from the arXiv website. Your task is to evaluate whether this text contains elements of mathematical intelligence and if it is suitable for educational purposes for YOURSELF in the field of mathematics. Please respond with only YES or NO
</system>

User: {
  "Title": "{title}",
  "Abstract": "{abstract}",
  "Text": "{text}"
}
1. Does the text contain elements of mathematical intelligence? Reply with only YES or NO
2. Is the text suitable for educational purposes for YOURSELF in the field of mathematics? Reply with only YES or NO
"
Assistant: 1.

Figure 8: Prompt for selecting the papers from arXiv.org.

"
<system>
You are ChatGPT, the most capable large language model equipped with extensive expertise in mathematics and coding, particularly skilled in complex reasoning and problem-solving. In the following interaction, I will provide you with a code excerpt from a website. Your task is to evaluate whether this code contains elements of mathematical intelligence and if it is suitable for educational purposes for YOURSELF in the field of mathematics. Please respond with only YES or NO
</system>

User: {
  "url": "{url}",
  "text": "{text}"
}
1. Does the code contain elements of mathematical intelligence? Reply with only YES or NO
2. Is the code suitable for educational purposes for YOURSELF in the field of mathematics? Reply with only YES or NO
"
Assistant: 1.

Figure 9: Prompt for selecting code snippets from GitHub.

B.2 Alternative Score functions

One can use alternative scoring functions corresponding to different partition functions, such as the formulas shown below.

$$\text{LM-Score}_{\text{alternative}}(\cdot) = \frac{\exp(\max(\text{logit}(\text{`YES'}), \text{logit}(\text{`Yes'})))}{\exp(\max(\text{logit}(\text{`YES'}), \text{logit}(\text{`Yes'}))) + \exp(\max(\text{logit}(\text{`NO'}), \text{logit}(\text{`No'})))} \tag{5}$$

Or:

$$\text{LM-Score}_{\text{alternative-II}}(\cdot) = \frac{\exp(\text{logit}(\text{`YES'})) + \exp(\text{logit}(\text{`Yes'}))}{\exp(\text{logit}(\text{`YES'})) + \exp(\text{logit}(\text{`Yes'})) + \exp(\text{logit}(\text{`NO'})) + \exp(\text{logit}(\text{`No'}))} \tag{6}$$