Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

Exploring Advanced Reasoning Abilities of Large Language Models in Slovak

Bachelor Thesis

2024
Adam Zahradník

Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

Exploring Advanced Reasoning Abilities of Large Language Models in Slovak

Bachelor Thesis

Study Programme:	Applied Computer Science
Field of Study:	Computer Science
Department:	Department of Applied Informatics
Supervisor:	Mgr. Marek Šuppa

Bratislava, 2024
Adam Zahradník

48302124

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Adam Zahradník
Študijný program: aplikovaná informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: informatika
Typ záverečnej práce: bakalárska
Jazyk záverečnej práce: anglický
Sekundárny jazyk: slovenský

Názov: Exploring Advanced Reasoning Abilities of Large Language Models in Slovak
Skúmanie pokročilých schopností uvažovania veľkých jazykových modelov v slovenčine

Anotácia: V nedávnej minulosti veľké jazykové modely (LLM) preukázali pôsobivé schopnosti pri riešení rôznych kvantitatívnych úloh a úloh v oblasti vedomostí v oblastiach, ako je matematika, fyzika a informatika. Tieto úspechy sa zvyčajne hodnotia pomocou metodológií navrhnutých pre základné a stredné školy, často odvodených zo štandardizovaných testov. Tento prístup však predstavuje problém: štandardizované testy sa často dostanú do tréningových datasetov LLM, čo vedie k skresleným hodnoteniam ich výkonu. Okrem toho hodnotenie rozumových schopností (či schopnosti uvažovania) LLM prebieha primárne v angličtine, čo vyvoláva obavy z použiteľnosti a zovšeobecniteľnosti výsledkov týchto hodnotení.

Cieľ: Ciele bakalárskej práce zahŕňajú (ale nie sú obmedzené na)
- analýza súčasného stavu hodnotení schopnosti uvažovania
- vytváranie a/alebo zhromažďovanie logických hodnotiacich datasetov v slovenskom jazyku na základe úloh/cvičení z matematických, fyzikálnych alebo informačných olympiád, ako aj rôznych korešpondenčných seminárov z rovnakých oblastí
- vyhodnotenie najmodernejších LLM na pripravených datasetoch
- analýza výstupov (a chybových režimov) vytvorených modelmi s najlepšími výsledkami

Literatúra:
Hendrycks, Dan, et al. "Measuring mathematical problem solving with the math dataset." arXiv preprint arXiv:2103.03874 (2021). (https://arxiv.org/pdf/2103.03874.pdf)
Cobbe, Karl, et al. "Training verifiers to solve math word problems." arXiv preprint arXiv:2110.14168 (2021). (https://arxiv.org/pdf/2110.14168.pdf)
Wei, Jason, et al. "Chain-of-thought prompting elicits reasoning in large language models." Advances in Neural Information Processing Systems 35 (2022): 24824-24837. (https://arxiv.org/abs/2201.11903)
Sawada, Tomohiro, et al. "Arb: Advanced reasoning benchmark for large language models." arXiv preprint arXiv:2307.13692 (2023). (https://arxiv.org/pdf/2307.13692.pdf)

Vedúci: Mgr. Marek Šuppa
Katedra: FMFI.KAI - Katedra aplikovanej informatiky

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Vedúci katedry: doc. RNDr. Tatiana Jajcayová, PhD.

Dátum zadania: 15.10.2023

Dátum schválenia: 16.10.2023 doc. RNDr. Damas Gruska, PhD.
garant študijného programu

.....
študent

.....
vedúci práce

Comenius University Bratislava
Faculty of Mathematics, Physics and Informatics

THESIS ASSIGNMENT

Name and Surname: Adam Zahradník
Study programme: Applied Computer Science (Single degree study, bachelor I. deg., full time form)
Field of Study: Computer Science
Type of Thesis: Bachelor´s thesis
Language of Thesis: English
Secondary language: Slovak

Title: Exploring Advanced Reasoning Abilities of Large Language Models in Slovak

Annotation: In the recent past, Large Language Models (LLMs) have shown impressive capabilities in tackling various quantitative reasoning and knowledge challenges in fields like mathematics, physics, and computer science. These achievements are typically assessed using benchmarks designed for primary and secondary school levels, often derived from standardized tests. However, this approach poses a problem: standardized tests frequently find their way into the training data of LLMs, leading to skewed performance evaluations. Additionally, the evaluation of LLMs' reasoning abilities primarily occurs in English, raising concerns about the applicability and generalizability of the results.

Aim: The goals of the bachelor's thesis include (but are not limited to)
- analysis of the current state-of-the-art in reasoning capability evaluation
- creation and/or collection of reasoning evaluation datasets in Slovak language based on tasks/exercises in Olympiads in Mathematics, Physics or Informations, as well as various correspondence seminars in the same areas
- evaluation of state-of-the-art LLMs on the prepared datasets
- analysis of the outputs (and error modes) produced by the best performing models

Literature:
Hendrycks, Dan, et al. "Measuring mathematical problem solving with the math dataset." arXiv preprint arXiv:2103.03874 (2021). (https://arxiv.org/pdf/2103.03874.pdf)
Cobbe, Karl, et al. "Training verifiers to solve math word problems." arXiv preprint arXiv:2110.14168 (2021). (https://arxiv.org/pdf/2110.14168.pdf)
Wei, Jason, et al. "Chain-of-thought prompting elicits reasoning in large language models." Advances in Neural Information Processing Systems 35 (2022): 24824-24837. (https://arxiv.org/abs/2201.11903)
Sawada, Tomohiro, et al. "Arb: Advanced reasoning benchmark for large language models." arXiv preprint arXiv:2307.13692 (2023). (https://arxiv.org/pdf/2307.13692.pdf)

Supervisor: Mgr. Marek Šuppa
Department: FMFI.KAI - Department of Applied Informatics
Head of department: doc. RNDr. Tatiana Jajcayová, PhD.

Comenius University Bratislava
Faculty of Mathematics, Physics and Informatics

Assigned: 15.10.2023

Approved: 16.10.2023 doc. RNDr. Damas Gruska, PhD.
Guarantor of Study Programme

Student

Supervisor

Acknowledgments: Tu môžete poďakovať školiteľovi, prípadne ďalším osobám, ktoré vám s prácou nejako pomohli, poradili, poskytli dáta a podobne.

Abstrakt

Slovenský abstrakt v rozsahu 100-500 slov, jeden odstavec. Abstrakt stručne sumarizuje výsledky práce. Mal by byť pochopiteľný pre bežného informatika. Nemal by teda využívať skratky, termíny alebo označenie zavedené v práci, okrem tých, ktoré sú všeobecne známe.

Kľúčové slová: jedno, druhé, tretie (prípadne štvrté, piate)

Abstract

Abstract in the English language (translation of the abstract in the Slovak language).

Keywords:

Contents

Chapter 1

Introduction

Latest developments in the field of artificial intelligence led to great improvements in the abilities of large language models to solve many different types of tasks. Prior work demonstrates ambitious results on tasks that challenge the models' ability to reason about maths, physics and informatics. Researchers introduced many datasets and methods to evaluate the large language models' abilities to solve these problems.

These datasets and methods are usually based off standardized elementary and high school tests in these fields. Standardized tests that are publicly available can end up included in the models' training data, which can bias the models' evaluation results. Currently, most available datasets and evaluation methods are in English, which raises the question whether observed large language models' abilities can be generalized to other languages as well.

1.1 Large Language Models

Large language models are a recent advancement in artificial intelligence, particularly in the realm of generative models. They operate by first receiving an initial input text, which is also called a prompt. Then, the model uses its neural network to predict the next word or token. This token is then appended to the prompt, creating an extended text input. The process is repeated until a specified length of text or another stopping criterion is reached. The result is a coherent and contextually relevant piece of text, generated entirely by the model's learned patterns and associations within its training data.

The prompt that was given to the large language model together with the generated text is called a context window. The model can recall information from its context window. This allows the user to provide additional context to the model. Alternatively, the model can in some circumstances use its context window as a kind of scratch pad and thus better prepare its output. This is taken advantage of by some of the prompting

techniques.

It was observed that large language models exhibit common-sense reasoning capabilities [14]. More importantly, these models can perform advanced reasoning needed to solve mathematical problems [12]. Even though large language models are often less capable than humans in solving such problems, it still allows them to be used in a wide range of applications.

1.2 Prompting Techniques

Prior research has shown that manipulating the way in which the model's prompt is constructed can have a significant impact on the quality of the resulting output. In this section, we introduce successful techniques that we use to compare large language model performance in the Slovak language.

1.2.1 Zero-, One- and Few-shot Prompting

Even though large language models are trained on generic text corpus datasets, prior research has indicated that LLMs do not require fine-tuning the model on the desired task [15][3].

Brown et al. [3] shows that this fine-tuning step can be replaced by a technique called few-shot prompting. Few-shot prompting provides the model with few examples of the desired task, along with sample solutions right in the model's input. This measurably improves the models' capability to solve the desired task, even though the model was not trained to solve that particular task in the first place. Few-shot prompting is typically done with 10 to 100 task examples, depending on the size of the model's context window.

There are two related techniques to few-shot prompting, introduced by Brown et al. [3]: one-shot and zero-shot prompting. One-shot prompting is done in the same way as few-shot, but the model is provided with only one example of the task. Zero-shot prompting is a similar, but generally less effective, technique in which the model is provided with a natural language description of the task instead of any examples.

1.2.2 Chain-of-Thought

Another promising prompting technique is chain-of-thought introduced by Wei et al. [21]. This technique mimics one's own thought process when solving tasks. The goal of chain-of-thought prompting is to make the model generate a series of intermediate steps that lead to the final answer of a problem. Wei et al. [21] shows that large language models are capable of generating such chain-of-thoughts when provided with such chain-

of-thoughts in the examples used for few-shot prompting. The model is provided with example solutions that walk the reader through the different steps leading to the final answer.

1.2.3 Zero-shot Chain-of-Thought

Classical chain-of-thought prompting as introduced by Wei et al. [21] has the disadvantage of needing to provide examples of task solutions including chain-of-thoughts, which are usually not readily available.

To address such problems, Kojima et al. [6] introduced a simple, yet effective technique, called zero-shot chain-of-thought. The idea is that when the model is prompted to "think step by step", it can generate a chain-of-though without needing any examples beforehand. So, the model is prompted with the question and a simple prompt like "Let's think step by step" to force it to generate a chain-of-thought.

1.2.4 Generated Knowledge

Another similar approach to zero-shot chain-of-thought was demonstrated by Liu et al. [10]. The generated knowledge prompting technique leverages the fact that large language models can use their context windows for short-term memory. This is very similar to some teaching techniques employed when teaching humans new concepts.

The model is prompted to first describe all concepts relevant to solving the problem and then attempt to solve the problem. This way, the model will introduce a lot of new information into its context window. It can later retrieve information from the context window to help itself to solve the problem.

1.2.5 Dual Prompt Generated Knowledge

The generated knowledge prompting method has a slight disadvantage - the model has a limited number of words or tokens it can produce. Because of this, it can spend a lot of its available space on preparing the relevant context and end up not having enough tokens left for the solution itself.

Dual prompt generated knowledge improves on this by prompting the model to only generate the relevant context. After it generates the context, the model is prompted again with the original question and the context it generated. This allows the model to generate longer answers.

1.2.6 Least-to-Most Prompting

Least-to-most prompting takes advantage of the model's context window combined with multiple prompts. The idea introduced by Zhou et al. [22] is that we break the problem into smaller sub-problems, which are then solved sequentially.

We start by prompting the model with the problem and ask it to list out the sub-problems that are required to solve the whole problem. We then take the first sub-problem it generates and ask it to solve it. This is usually done by prompting it with the original problem and a question to solve a given sub-problem. The solution of the sub-problem is then appended to the prompt along with another sub-problem. This process is repeated until the model solves all sub-problems. At that point, we should have the whole solution.

1.3 Existing Datasets

Large language models have already been evaluated on mathematical reasoning tasks by researchers using numerous datasets. Most of these datasets were created by scraping problems from the internet. We provide a comparison of few selected datasets related to our work to better understand the types and problems involving creating such a dataset.

MultiArith released by Roy and Roth [16] contains multistep arithmetic problems without irrelevant quantities. That means that those problems require a combination of different arithmetic operations to get the answer and don't have any information that isn't needed to solve the problem. The dataset contains symbolic solutions.

Math23K by Wang et al. [20] consists of Chinese elementary school maths problems scraped from the internet. This dataset contains only problems with single linear unknown variable. This dataset contains only symbolic solutions.

AQuA introduced by Ling et al. [9] consists of multi-choice word problems covering a broad range of topics and difficulty levels. This dataset also contains descriptions of the rationale to reach the correct answer.

MATH is a dataset consisting of challenging competition mathematic problems with step-by-step natural language solutions introduced by Hendrycks et al. [5]. The problems were retrieved from United States' mathematics competitions. These problems are designed to be challenging for humans and often require more than just basic application of mathematic tools.

GSM8K released by Cobbe et al. [4] consists of multistep middle school word problems with natural language solutions. These problems take between 2 and 8 steps to solve. A bright student should be able to solve all of them.

ASDiv is a mathematical word problem dataset with a strong emphasis on great

diversity. This dataset of arithmetic and algebraic problems was introduced by Miao et al. [11].

SVAMP introduced by Patel et al. [13] contains many variations of elementary school mathematical word problems.

MGSM is a multilingual dataset introduced by Shi et al. [19] containing 250 manually translated grade-school problems from the GSM8K.

Dataset	Size	Answer	Difficulty	Language
MultiArith	600	symbolic	elementary school	English
Math23K	23 161	symbolic	elementary school	Chinese
AQuA	100 949	multi-choice	diverse	English
MATH	12 500	natural language	competitions	English
GSM8K	8 500	natural language	elementary school	English
ASDiv	2 305	symbolic	elementary school	English
SVAMP	1 000	symbolic	elementary school	English
MGSM	250	natural language	elementary school	multilingual

Figure 1.1: Comparison of existing datasets

1.4 Evaluating LLM Answers

There are few ways to evaluate answers generated by LLMs. The method used varies depending on the type of question. We will only focus on problems that have known solutions that can be used to verify the model's answers.

For questions with concrete numerical answers, the most straightforward approach is to compare the LLM's numerical answer to the correct one. However, LLMs usually generate natural language output. This usually involves extracting the number or equation from the model's output. Such an approach is very precise, as shown by Hendrycks et al. [5], Wang et al. [20] and Sawada et al. [17].

LLMs can also be evaluated on multiple-choice problems. This is done by providing the model with the options and prompting it to select one of the provided options [19]. With the right prompting, models can output answers that can be extracted in more than 97% of the time [17]. Alternatively, the model is not provided with the options, but its answer is extracted as a number and then compared to the available options as in Amini et al. [2].

The most problematic questions are those which have open answers that cannot be easily extracted from the model's output. Unfortunately, these are the types of questions that we are most interested in. One of the approaches is to try and convert these questions into ones that allow automatic answer extraction. This is done by extracting a number or equation from the reference solution and trying to match it

with the model's output. Other methods involve changing the question into a multiple-choice one. Both of those approaches, while valid, do not fully evaluate the model's capabilities to correctly solve problems that require natural language solutions.

The most straightforward approach to evaluating such problems is using a human evaluator. Nonetheless, this approach is highly labor-intensive and ineffective. We can leverage the existing model's capabilities and use it instead of a human to evaluate the answers. Such an approach is called model-based evaluation [17].

In its simplest form, the model is provided by the reference solution and the output that should be evaluated. It is then asked to grade the provided output. Prior research indicates that such an approach is possible, but the evaluations are not reliable enough to be used alone [7] [18]. Some researchers went so far as to avoid providing the model with the reference solution. Those experiments provided promising results, but they still failed to be reliable enough [8].

An improved approach was introduced by asking the model to generate evaluation rubrics, and then using those rubrics to evaluate the solutions [17]. The model is provided with the reference solution and generates rubrics and allocates points to them. It was shown by Sawada et al. [17] that GPT-4 designs rubrics that cover most of the solution steps correctly, but sometimes fail to properly allocate points based on their importance. The model is quite reliable on assigning the correct number of points to solutions based on the generated rubrics. However, the model cannot score solutions that do not follow the generated rubrics, but are otherwise correct. Another issue with this approach is that the model attempts to assign points to attempted solutions that are outside the generated rubrics. A human evaluator would score these solutions with zero points [17]. Knowing its limitations, we will base our approach on this method to evaluate models on our dataset.

1.5 Prior Research

The MGSM paper by Shi et al. [19] tries to evaluate models' reasoning abilities in multiple languages. They achieve this by manually translating 250 problems from GSM8K [4] into ten typologically diverse languages, which they then used to benchmark GPT-3 on their dataset. Their research reveals that results are very similar, with insignificant differences between the various languages. It has been demonstrated that using an intermediate English chain-of-thought provides results that are on par with or better than answers written in the question's original language.

Another multilingual research by Ahuja et al. [1] comprehensively evaluates the models on various multilingual datasets. Even though this research does not evaluate the advanced reasoning abilities, it demonstrates the overall capabilities of large

language models to reason in languages other than English. Their results show no significant differences between the results achieved in the different languages.

Bibliography

[1] Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Maxamed Axmed, Kalika Bali, and Sunayana Sitaram. Mega: Multilingual evaluation of generative ai, 2023.

[2] Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. *CoRR*, abs/1905.13319, 2019. URL `http://arxiv.org/abs/1905.13319`.

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

[4] 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. Training verifiers to solve math word problems, 2021.

[5] 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, 2021.

[6] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners, 2023.

[7] Gerd Kortemeyer. Toward AI grading of student problem solutions in introductory physics: A feasibility study. *Physical Review Physics Education Research*, 19(2), November 2023. ISSN 2469-9896. doi: 10.1103/physrevphyseducres.19.

020163. URL http://dx.doi.org/10.1103/PhysRevPhysEducRes.19.020163.

[8] Gerd Kortemeyer. Performance of the pre-trained large language model GPT-4 on automated short answer grading, 2023.

[9] Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1015. URL https://aclanthology.org/P17-1015.

[10] Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. Generated knowledge prompting for commonsense reasoning, 2022.

[11] Shen-Yun Miao, Chao-Chun Liang, and Keh-Yih Su. A diverse corpus for evaluating and developing english math word problem solvers, 2021.

[12] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger

Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024.

[13] Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are NLP models really able to solve simple math word problems?, 2021.

[14] Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. Is chatgpt a general-purpose natural language processing task solver?, 2023.

[15] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. URL https://api.semanticscholar.org/CorpusID:160025533.

[16] Subhro Roy and Dan Roth. Solving general arithmetic word problems, 2016.

[17] Tomohiro Sawada, Daniel Paleka, Alexander Havrilla, Pranav Tadepalli, Paula Vidas, Alexander Kranias, John J. Nay, Kshitij Gupta, and Aran Komatsuzaki. ARB: Advanced reasoning benchmark for large language models, 2023.

[18] Johannes Schneider, Bernd Schenk, Christina Niklaus, and Michaelis Vlachos. Towards LLM-based autograding for short textual answers, 2023.

[19] Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. Language models are multilingual chain-of-thought reasoners, 2022.

[20] Yan Wang, Xiaojiang Liu, and Shuming Shi. Deep neural solver for math word problems. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 845–854, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1088. URL https://aclanthology.org/D17-1088.

[21] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023.

[22] Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. Least-to-most prompting enables complex reasoning in large language models, 2023.