Uncovering the Intention Behind Equations in Mathematical Problems

Anonymous EMNLP submission

Abstract

Mathematical Equation Intent Recognition(MEIR) is a novel task aimed at identifying the intentions behind mathematical equations that people produce while solving **math world problems(MWPs)**. We observe that, in previous research, researchers have often focused on how to let **large language models(LLMs)** correctly solve an MWP. However, focusing solely on the reasoning behind each step of a correct inference process is insufficient. We prefer that LLMs can provide guidance on the process of solving MWPs for students in educational settings. Therefore, they need to adjust the strategy based on the student's responses. We notice that, unlike existing mathematical datasets, students typically do not provide overly detailed descriptions of their steps in the real world. As a result, it is crucial for LLMs to possess the capability to understand the intention they produce those equations. We treat MEIR as a generation task, requiring models to summarize the intent in a single sentence. We also propose a data augmentation framework and utilized this framework to generate a benchmark called **Grade School Math Intention(GSMI)**. To evaluate MEIR task, we benchmark serveral LLMs on GSMI dataset. The results indicate that there is still significant room for improvement in the performance of general-purpose LLMs on the MEIR task. Conversely, capabilities acquired during pre-training and fine-tuning specifically in the field of mathematics significantly contribute to the model's ability to tackle those problems. Codes and datasets are available on https://github.com/ch-666-six/MEIR

1 Introduction

Recently, the capabilities of large language models(LLMs) (Minaee et al., 2024) have been extensively applied to tasks in the field of mathematics. Numerous researchers (Liu et al., 2023c; Wei et al., 2022; Kojima et al., 2022) have employed prompt-based methods or fine-tuning methods to further enhance the ability of large language models to comprehend mathematical texts and solve mathematical problems.

Question: The price of a laptop is $1000. If you get a 20% discount, how much do you have to pay?

Complete Answer(From GSM8k):

You will get a discount of **20/100 * $1000 = $<<20/100*1000=200>>200.**

Therefore, you will have to pay **$1000 - $200 = $<<1000-200=800>>800.**

So the answer is: 800.

Brief Answer(From Student):

According to the question, the solution process of this problem is as follows:

1000-20/100*100=800.

As a result, we should pay 800 dollars.

Figure 1: An example of complete answer and brief answer. The standard answer is sourced from the GSM8K dataset(Cobbe et al., 2021), reflecting the ideal scenario of solving mathematical problems. However, in real-time scenarios, answers from students may resemble what is shown in the "brief answer".This answer may primarily consists of a series of mathematical equations.

However, we do not want LLMs simply become problem solvers. We hope to integrate the LLMs' mathematical capabilities closely with real-world educational scenarios. We find that, in real-time educational scenarios, particularly during homework or exams, students often arrive at their answers through a series of equations rather than a detailed step-by-step reasoning process. We show an example in Figure 1, citing a math question from GSM8k dataset(Cobbe et al., 2021).

As a result, to determine the correctness of the problem-solving process, we need to fully understand the intent behind these equations. Therefore, it is important to study the ability of LLMs to understand the intent behind the arithmetic equations.

We consider the task of **Mathematical Equation Intent Recognition(MEIR)** as a generation task. Specifically, we aim for the language model to produce concise descriptions for the equations within the solution steps of mathematical problems. To address this issue, relevant data is of necessity.

We use two different modules: Imitation-based Generator and Intention Extractor to generate data automatically, and propose a novel dataset called **Math World Problems Intention(MWPI)**. Details will be thoroughly explained in section 3.

MWPI is a benchmark to test whether language models can uncover the intention behind those equations appeared in the solutions of math world problems. The input context consists of a mathematical problem along with its solution steps expressed in the form of equations. The objective of the model is to produce, for each equation, a concise summary in the form of a sentence that encapsulates the intention behind the inclusion of that particular equation.

In our experimental evaluation, we observed that mainstream closed-source large language models (LLMs), such as GPT-4o, GPT-4 (OpenAI, 2024) and others, still exhibit potential for improvement in the MEIR task. This suggests that during the pre-training process, these models did not systematically acquire the ability to parse mathematical equations. In addition, we selected several open-source models and utilized instruction tuning to train them in the process of parsing mathematical expressions. We demonstrate that through specific instruction tuning, models with smaller parameter sizes can also achieve good performance. Meanwhile, through imitation-based generator, language model can improve themselves sustainably.

To conclude, the main contributions of this article are as follows:

1. To the best of our knowledge, we are the first to explore MEIR task.
2. We introduce a novel dataset called MWPI to evaluate the performance of models on MEIR task.
3. We employ an imitation-based generator to facilitate the generation of more diverse data under limited resources.

2 Related Work

2.1 Math World Problems Solving

Large language model have a strong ability to solve math world problems. Chain-of-thought prompting(Wei et al., 2022; Zhang et al., 2023b; Kojima et al., 2022) is a highly effective technique for eliciting detailed reasoning processes from LLMs to solve mathematical problems. Some researchers(Liu et al., 2023b; Gou et al., 2023; Imani et al., 2023) also utilize external tools, like python executor and mathematical calculator, to enhance the calculate abilities of LLMs to solve mathematical problems. In addition, some researchers(Yu et al., 2023; Luo et al., 2023a; Ho et al., 2023; An et al., 2023) have organized mathematical corpora and fine-tuned open-source models using these corpora to enhance the mathematical reasoning capabilities. On several benchmark datasets (Cobbe et al., 2021; Hendrycks et al., 2021), LLMs have already demonstrated outstanding performance.

2.2 Instruction Tuning

Instruction tuning (Zhang et al., 2024) is an essential method for improving the capabilities and controllability of LLMs. This approach uses (INSTRUCTION, OUTPUT) pairs to train LLMs, where INSTRUCTION represents human instruction and OUTPUT denotes the target output that follows the instruction. LLMs like InstructGPT(Ouyang et al., 2022), Flan-T5(Chung et al., 2022), WizardLM(Xu et al., 2023), LLAVA(Liu et al., 2023a) and so on, are trained through instruction tuning. In domain-specific settings(Gupta et al., 2022; Zhang et al., 2023a; Luo et al., 2023b; Liu and Low, 2023), instruction tuning can also have a profound impact and contribute significantly to the performance. Compared to standard LLMs, instruction tuning enables more controllable and predictable model behavior. Due to the significant advantages of instruction fine-tuning, we also employed instruction tuning methods in our research.

2.3 Intent Understanding

Intent understanding(Louvan and Magnini, 2020) is one of the crucial tasks in artificial intelligence. In human-computer interaction(Jaimes and Sebe, 2007), accurately recognizing human intent facilitates machines in taking more appropriate actions to provide feedback. For example, in online shopping(Rahman et al., 2024; Yu et al., 2024), merchants always want to understand and accurately predict the buyer's intention to promote consumption. Some researchers(Yin et al., 2024; Weld et al., 2022; Hariharan et al., 2022) treat intent understanding as intent classification and slot filling tasks. By contrast, in order to fully understand students' intentions behind the equations, we view intent understanding as a text generation task(Li et al., 2021).

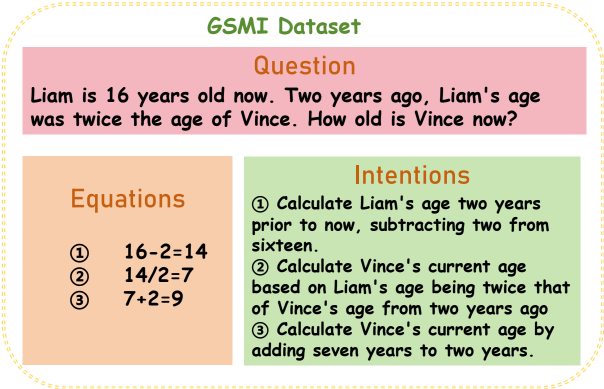

Figure 2: An example of GSMI dataset. Each data comprises a mathematical problem, a set of equations solving that problem and intention descriptions corresponding to each equation.

3 Dataset Construction

3.1 Overview

We propose a novel dataset called **Math World Problems Intention(MWPI)**. Each data in the dataset consists of 3 components: Question, Equations and Intentions. We provide an example in Figure 2 to illustrate the structure of this dataset.

For clarity in intent representation, we establish the rule that each sentence must begin with the word "calculate" and contain no more than 30 words.

Building on the existing dataset like GSM8k(Cobbe et al., 2021) and MATH(Hendrycks et al., 2021), we adopt the following two modules to generate the GSMI dataset: an imitation-based generator and an intention extractor. Figure 3 shows the following process.

Through the imitation-based generator the intention extractor, we can generate more valid instances to evaluate MEIR task. We utilize Chatgpt and GPT-4o(OpenAI, 2024) as LLMs to construct this two modules. The most labor-intensive step in this process is verifying the correctness of the expanded mathematical problems generated by LLMs. In this version, the GSMI dataset contains 8K training samples and 600 testing samples.

3.2 Imitation-based Generator

In the MEIR task, we focus on arithmetic problems of elementary school difficulty and in text modality. To enhance the model's ability to learn the extraction of mathematical expression intentions, we implement data augmentation techniques(Li et al., 2022; Zhou et al., 2024).

Algorithm 1 Imitation-based Generator

Input: Original Question Dataset S
Large Language Model $LLM()$
Input Prompt $P()$
Output: Expanded Question Dataset S'

```
S'=[]
while Normal Execution do
    Q = Random_sample(S)
    Q'= LLM(P(Q))
    if Grammar_Error(Q') then
        CONTINUE
    end if
    if Answer_Error(Q') then
        CONTINUE
    end if
    S' = S' + [Q']
end while
return S'
```

Motivated by (Wei et al., 2022), we conclude that large language models possess significant in-context learning capabilities(Dong et al., 2023; Li, 2023). In the Chain-of-Thought(CoT) prompting method, researchers provide a step-by-step reasoning example within the input prompt. Guided by this example, LLMs like GPT-4(OpenAI, 2024) can mimic the provided instance from the prompt to perform structured reasoning process on a new mathematical problem.

Similarly, we innovatively propose the concept of an imitation-based generator. In our approach, we present a mathematical problem along with its corresponding solution process in the input prompt, instructing the LLMs to imitate the contextual information and generate a new problem that is structurally similar and of comparable difficulty. In this process, we utilize text-only ChatGPT to generate problems. The corresponding algorithm is shown in the Algorithm 1.

3.3 Intention Extractor

The purpose of this module is to extract the intent within mathematical equations. Firstly, for each step in the chain of thought, we extracted the mathematical equations representing that step. Next, we used the textual information of each step as the input prompt, allowing the large language model to summarize the intention within each step. The corresponding algorithm is shown in the Algorithm 2.

In short, after employing the aforementioned

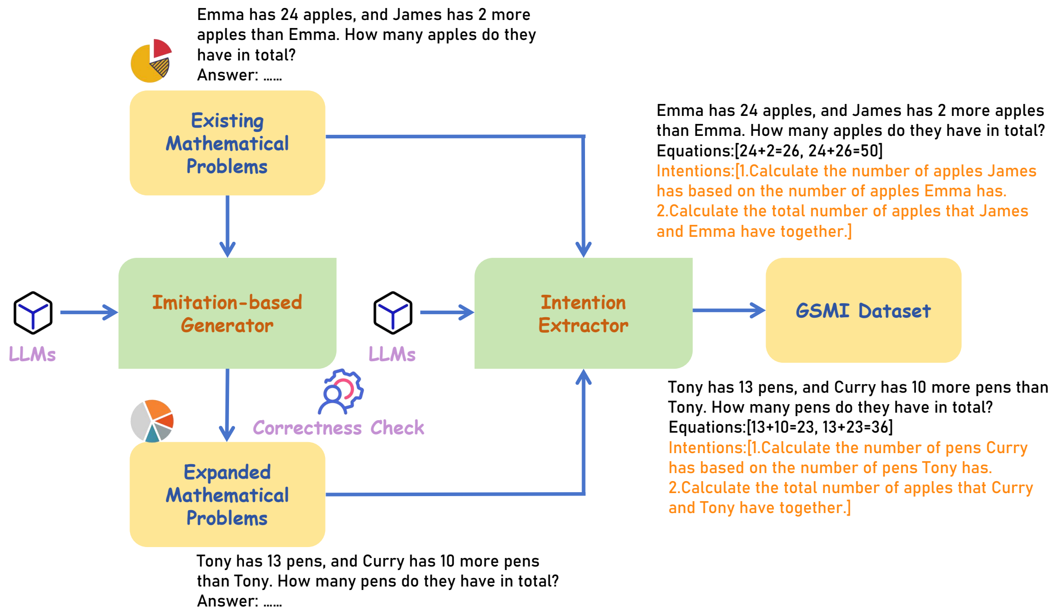

Figure 3: The Process of building GSMI dataset. The Imitation-based Generator is used to expand existing mathematical problems. The Intention Extractor is used to extract the intentions or objectives within each mathematical step. The figure illustrates a simple example from the GSMI dataset.

Algorithm 2 Intention Extractor

Input: Reasoning Step Set $R[\,]$
Large Language Model $LLM()$
Input Prompt $P()$
Output: Intention Set $I[\,]$
I = []
for each item r in R **do**
  i=LLM(P(r))
  $I = I$ + [i]
end for
return I

two modules, we can continuously generate data in GSMI, facilitating the model's improved learning of the recognition of mathematical expression intents.

4 Experiments

4.1 Experimental Setup

We use the GSMI dataset to evaluate the model's capability in recognizing the intent of mathematical expressions. For this purpose, we selected the following candidate models.

- **GPT-4o**(OpenAI, 2024) is a multilingual, multimodal generative pre-trained transformer designed by OpenAI. It was announced on 13 May, 2024, and released in the same day.
- **GPT-4** (OpenAI, 2024) is also a generative model designed by OpenAI, and it was announced in March, 2023.
- **GPT-3.5** (OpenAI, 2024), also known as ChatGpt, is a powerful large-scale language model. It was announced by OpenAI in March, 2022.
- **LLaMA** (Touvron et al., 2023) is a family of autoregressive large language models released by Meta AI, and we use the LLaMA-2 and LLaMA-3 models.
- **MetaMath** (Yu et al., 2023) is a fine-tuned model specifically for the field of mathematics. Researchers fine tune LLaMA (Touvron et al., 2023) on MetaMathQA dataset(Yu et al., 2023) and obtain MataMath.
- **WizardMath** (Luo et al., 2023a) is a fine-tuned model for mathematics. Researchers train WizardMath model using reinforcement learning methods.

For open-source models, we performed instruction fine-tuning using the training dataset and then evaluated the models using the testing dataset. Due to the constraints on computational resources, we adopted the LoRA (Low-Rank Adaptation) parameter-efficient fine-tuning approach(Hu et al., 2021). By default, the open-source model is trained

MODEL	BLEU-1	ROUGE-L			BERT-SCORE		
		P	R	F1	P	R	F1
Prompting Closed-source Models							
GPT-4o	0.2278	0.5691	0.4533	0.4944	0.8121	0.7844	0.7971
GPT-4	0.2194	0.5826	0.3879	0.4570	0.8098	0.7601	0.7834
GPT-3.5	0.2304	0.5210	0.4819	0.4910	0.7972	0.7942	0.7949
Tuning Open-source Models							
LLAMA-2-7b	0.2335	0.5206	0.4935	0.5001	0.8015	0.7919	0.7961
LLAMA-2-13b	**0.2386**	0.5450	0.5306	0.5316	0.8129	0.8095	0.8107
LLAMA-3-8b	0.2373	0.5436	0.5202	0.5252	0.8087	0.8061	0.8068
LLAMA-3-8b-instruct	0.2376	0.5416	0.5235	0.5261	0.8097	0.8069	0.8077
MetaMath-7b	0.2377	0.5392	0.5233	0.5255	0.8111	0.8079	0.8089
MetaMath-13b	0.2369	**0.5602**	0.5308	0.5386	**0.8186**	0.8105	**0.8139**
WizardMath-7b	0.2372	0.5549	**0.5400**	**0.5412**	0.8152	**0.8134**	0.8138

Table 1: Results on MEIR task. P means Precision. R means Recall. And F1 means F1 score. In the closed-source models, the best-performing value in each row is highlighted in yellow. In the open-source models, the best-performing value in each row is highlighted in green. The best value in each row is highlighted in bold.

on the training set for 3 epochs with a learning rate of 2e-4.

For closed-source models, we directly evaluated them using the testing dataset.

4.2 Evaluation metrics

We consider MEIR to be a text generation task. For the results generated by our candidate models for each equation, we need to evaluate their similarity to the ground truth. To this end, we selected the following evaluation metrics.

- **BLEU** (Papineni et al., 2002) is a metric for evaluating the quality of machine-generated text, which calculates precision for various n-gram lengths and combines these using a weighted geometric mean.

- **ROUGE** (Lin, 2004) is a set of metrics used to evaluate the quality machine-generated text. We use ROUGE-L, which captures the longest sequence of words that appear in both the candidate and reference summaries in the same order.

- **BERT-SCORE** (Zhang et al., 2020) leverages contextual embeddings from pre-trained transformer models, specifically BERT(Devlin et al., 2019) or RoBerta(Liu et al., 2019), to capture semantic similarity between the candidate and reference texts. We use bert-large-uncased as our base model, which contains 24 layers.

4.3 Experiment Result

The experimental results of the MEIR task are shown in the Table 1. We meticulously recorded the performance of all candidate models, retaining four decimal places, and documented the results in the table.

We observed that in the closed-source models, GPT-4o(OpenAI, 2024), as the latest proposed model, performs the best on the ROUGE-L and BERT-SCORE metrics. This indicates that GPT-4o surpasses the previously proposed GPT-4 and GPT-3.5 models in executing the MEIR task.

However, it is important to note that as a general-purpose large language model, GPT-4o, along with other closed-source models, lacks sufficient pre-training in the field of mathematics. As seen in the table, models with relatively smaller parameter sizes can outperform general-purpose large language models on the MEIR task after undergoing instruction tuning.

5 Analysis and Discussion

5.1 Mathematical Fine-tuning

MetaMath(Yu et al., 2023) and WizardMath(Luo et al., 2023a) are both fine-tuned versions of the LLAMA model. They were fine-tuned using extensive mathematical data, for example, the Meta-

Question	Chenny is 10 years old. Alyana is 4 years younger than Chenny. How old is Anne if she is 2 years older than Alyana?
Equations	["10-4=6", "6+2=8"]
GPT-4o Results	**(1) Calculate how much younger Alyana is than Chenny. (2) Calculate how much older Anne is than Alyana.**
MetaMath-13b Results	**(1) Calculate Alyana's age by subtracting four from ten. (2) Calculate Anne's age by adding two to six.**
WizardMath-7b Results	**(1) Calculate the age of Alyana by subtracting her age from Chenny's age.(2) Calculate Anne's age by adding six and two.**

Table 2: An simple example in GSMI testing set. In this example, GPT-4o clearly misunderstood the intent of the intermediate steps and provided an incorrect answer. MetaMath-13b and WizardMath-7b accurately grasped the intent of the intermediate steps.

MathQA dataset reached a size of 395K. As shown in Table 1, compared to LLAMA, MetaMath and WizardMath exhibit a significant advantage in handling the MEIR task. Notably, on the BERT-SCORE metric, which closely aligns with human evaluation, both models demonstrate remarkable capability.

Through controlled experiments, we have concluded that: **Mathematical Fine-tuning is highly effective and necessary for downstream mathematical tasks.**

5.2 Model Size

With LLMs demonstrating powerful capabilities across various domains, many people have begun to believe that there is a positive correlation between the parameter size of a model and its ability to handle complex problems.

However, as shown in the Table 1, on the MEIR task, the performance of smaller open-source models surpasses that of larger closed-source models. This indicates that for the MEIR task, **high-quality data refinement is more crucial than larger model sizes.** We require models to acquire knowledge and capability within a specific domain.

Table 2 presents an example from the evaluation set. In this instance, GPT-4o made evident errors in summarizing the intent, whereas MetaMath and WizardMath accurately summarized the intent.

5.3 Data Augmentation

In machine learning, richer datasets often yield better results, while a lack of data can easily lead to overfitting on the training data.

As shown in Figure 3, through Imitation-based Generator and Intention Extractor, we can continuously generate new data to further train the model on the MEIR task. Compared to collecting mathematical problems and answers from the real world, the method illustrated in Figure 3 clearly requires significantly less human effort and time.

However, we need to investigate the effectiveness of this data augmentation method. We raise a question that does generating more examples through Imitation-based Generator and Intention Extractor on the existing datasets enable the model to perform better on the MEIR task?

In this regard, we introduce a variable K. K represents the total number of examples involved in instruction tuning. We select K values of 500, 1000, 2000, and 5000 for experimentation on MetaMath and WizardMath models. The experimental results are shown in Table 3.

It is evident that as K increases, both ROUGE-L and BERT-SCORE metrics show an overall increase trend. When all data generated through Imitation-based Generator and Intention Extractor modules is used in the instruction tuning process, the performance also imporves significantly. These two modules can continuously generate new data. This indicates that we can leverage the imitation generation capability of LLMs to produce richer training data with limited resources. This part of training data truly helps language models better acquire the ability to uncover the intentions behind mathematical equations.

In summary, we state that **appropriate data augmentation strategies contributes to enhancing the language models' performance on the MEIR task.**

6 Conclusions

In this artical, we introduce the research efforts on unconvering the intention behind equations in

ROUGE-L F1 SCORE					
MODEL	K=500	K=1000	K=2000	K=5000	All Training Data
MetaMath-7B	0.5036	0.5197	0.5189	0.5246	0.5255
WizardMath-7B	**0.5241**	**0.5322**	**0.5408**	0.5316	**0.5412**
MetaMath-13B	0.5110	0.5201	0.5337	**0.5373**	0.5386
BERT-SCORE F1 SCORE					
MODEL	K=500	K=1000	K=2000	K=5000	All Training Data
MetaMath-7B	0.7979	0.8051	0.8053	0.8084	0.8089
WizardMath-7B	**0.8087**	**0.8117**	0.8126	0.8116	0.8138
MetaMath-13B	0.8008	0.8076	**0.8129**	**0.8132**	**0.8139**

Table 3: Results of the impact of generated data. The table records the performance of the model for different values of K. The maximum value in each row is highlighted with a pink shade, and the maximum value in each column is indicated in bold.

mathematical problems.

Firstly, we stated the importance of MEIR task. In real life, when handling mathematical problems, students might not provide very detailed descriptions for each step. However, the mathematical equations at each step are essential. Therefore, understanding the intention behind those listing equations at each step means comprehending the student's problem-solving approach. This is highly beneficial in the field of education.

Next, we introduced two modules: Imitation-based Generator and Intention Extractor. The Imitation-based Generator is used to increase data diversity. and the Intention Extractor is used to extract the intention behind each step. Through these two modules, we constructed the GSMI dataset. With minimal human resource consumption, these two modules can be used to generate more varied data. Experimental evidence has shown that the data generated by this structure is highly beneficial for improving model performance on MEIR tasks.

Subsequently, we selected a subset of candidate models and evaluated their ability to solve MEIR tasks on the GSMI dataset. The experimental results indicate that powerful general LLMs like GPT-4o still have shortcomings in understanding equations. Conversely, following a series of instruction tuning processes, small-scale open-source models demonstrate outstanding performance in understanding equations. Those models that have undergone mathematical fine-tuning, like MetaMath and WizardMath, excel in MEIR tasks.

In conclusion, we pioneered the study of equation intention analysis. We are the first to propose the MEIR task and have conducted thorough experiments to explore the capability of LLMs in addressing this task. Exploring equation intention is an interesting and important topic, and needs further attention and in-depth research.

7 Limitations and Future Works

In this experimental work, we have exposed certain limitations. Due to computational constraints, the maximum model parameter size we used for fine-tuning open-source models was 13 billion. In the future, we will run the MEIR task on larger-scale open-source models to explore their capabilities in understanding mathematical equations.

In our experiments, we have demonstrated that the data generated through these Imitation-based Generator and Intention Extractor modules helps improve the model's ability to understand equation intentions. In future work, we will propose more refined data augmentation mechanisms and introduce a larger-scale GSMI dataset.

Furthermore, for the generated data from Imitation-based Generator and Intention Extractor modules, we did not conduct comprehensive comparative analyses with existing datasets. In future work, an thorough comparative analysis is of necessity to make sure that our training data if of high quality.

Finally, as we have stated, the MEIR task closely aligns with educational settings. It is not sufficient to merely identify the intended meaning of correct equations. In the future, we aim to intelligently identify errors students make when producing equations in mathematical education scenarios. This places higher demands on language models, that they not only need to recognize and generalize the intended meaning of correct equations, but also need to uncover the underlying reasons for errors in incorrect equations.

References

Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2023. Learning from mistakes makes llm better reasoner.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. 2023. A survey on in-context learning.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Tora: A tool-integrated reasoning agent for mathematical problem solving. *CoRR*, abs/2309.17452.

Prakhar Gupta, Cathy Jiao, Yi-Ting Yeh, Shikib Mehri, Maxine Eskenazi, and Jeffrey Bigham. 2022. InstructDial: Improving zero and few-shot generalization in dialogue through instruction tuning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 505–525, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Shruthi Hariharan, Vignesh Kumar Krishnamurthy, Utkarsh, and Jayantha Gowda Sarapanahalli. 2022. Enhancing slot tagging with intent features for task oriented natural language understanding using bert.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. Large language models are reasoning teachers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 14852–14882. Association for Computational Linguistics.

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

Shima Imani, Liang Du, and Harsh Shrivastava. 2023. Mathprompter: Mathematical reasoning using large language models. In *Proceedings of the The 61st Annual Meeting of the Association for Computational Linguistics: Industry Track, ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 37–42. Association for Computational Linguistics.

Alejandro Jaimes and Nicu Sebe. 2007. Multimodal human–computer interaction: A survey. *Computer Vision and Image Understanding*, 108(1):116–134. Special Issue on Vision for Human-Computer Interaction.

Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213. Curran Associates, Inc.

Bohan Li, Yutai Hou, and Wanxiang Che. 2022. Data augmentation approaches in natural language processing: A survey. *AI Open*, 3:71–90.

Junyi Li, Tianyi Tang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. Pretrained language model for text generation: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4492–4499. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

Yinheng Li. 2023. A practical survey on zero-shot prompt design for in-context learning. In *Proceedings of the Conference Recent Advances in Natural Language Processing - Large Language Models for Natural Language Processings*, RANLP. INCOMA Ltd., Shoumen, BULGARIA.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916. Curran Associates, Inc.

Tengxiao Liu, Qipeng Guo, Yuqing Yang, Xiangkun Hu, Yue Zhang, Xipeng Qiu, and Zheng Zhang. 2023b. Plan, verify and switch: Integrated reasoning with diverse x-of-thoughts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 2807–2822. Association for Computational Linguistics.

Tiedong Liu and Bryan Kian Hsiang Low. 2023. Goat: Fine-tuned llama outperforms gpt-4 on arithmetic tasks.

Wentao Liu, Hanglei Hu, Jie Zhou, Yuyang Ding, Junsong Li, Jiayi Zeng, Mengliang He, Qin Chen, Bo Jiang, Aimin Zhou, and Liang He. 2023c. Mathematical language models: A survey.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Samuel Louvan and Bernardo Magnini. 2020. Recent neural methods on slot filling and intent classification for task-oriented dialogue systems: A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 480–496, Barcelona, Spain (Online). International Committee on Computational Linguistics.

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

Ziyang Luo, Can Xu, Pu Zhao, Qingfeng Sun, Xiubo Geng, Wenxiang Hu, Chongyang Tao, Jing Ma, Qingwei Lin, and Daxin Jiang. 2023b. Wizardcoder: Empowering code large language models with evol-instruct.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: A survey.

OpenAI. 2024. Gpt-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Muhammad Sabbir Rahman, Surajit Bag, Farhana Habib Zinnia, Nripendra P. Rana, and Mohammad Osman Gani. 2024. Understanding and predicting customers' intentions to use smartphone-based online games: A deep-learning-based dual-stage modelling analysis. *Computers in Human Behavior*, 152:108083.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Henry Weld, Xiaoqi Huang, Siqu Long, Josiah Poon, and Soyeon Caren Han. 2022. A survey of joint intent detection and slot filling models in natural language understanding. *ACM Comput. Surv.*, 55(8).

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions.

Shangjian Yin, Peijie Huang, and Yuhong Xu. 2024. Uni-mis: United multiple intent spoken language understanding via multi-view intent-slot interaction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19395–19403.

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

Wen-Ju Yu, Shin-Yuan Hung, Annie Pei-I Yu, and Yu-Li Hung. 2024. Understanding consumers' continuance intention of social shopping and social media participation: The perspective of friends on social media. *Information & Management*, 61(4):103808.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. 2024. Instruction tuning for large language models: A survey.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert.

Yue Zhang, Leyang Cui, Deng Cai, Xinting Huang, Tao Fang, and Wei Bi. 2023a. Multi-task instruction tuning of llama for specific scenarios: A preliminary study on writing assistance.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2023b. Automatic chain of thought prompting in large language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Yue Zhou, Chenlu Guo, Xu Wang, Yi Chang, and Yuan Wu. 2024. A survey on data augmentation in large model era.