John is 50 years old, can his son be 65?
Evaluating NLP Models' Understanding of Feasibility

Himanshu Gupta   Neeraj Varshney   Swaroop Mishra   Kuntal Kumar Pal
Saurabh Arjun Sawant   Kevin Scaria   Siddharth Goyal   Chitta Baral
Arizona State University

Abstract

In current NLP research, large-scale language models and their abilities are widely being discussed. Some recent works have also found notable failures of these models. Often these failure examples involve complex reasoning abilities. This work focuses on a simple commonsense ability, reasoning about when an action (or its effect) is feasible. To this end, we introduce FeasibilityQA, a question-answering dataset involving binary classification (BCQ) and multi-choice multi-correct questions (MCQ) that test understanding of feasibility. We show that even state-of-the-art models such as GPT-3, GPT-2, and T5 struggle to answer the feasibility questions correctly. Specifically, on MCQ and BCQ questions, GPT-3 achieves an accuracy of just (19%, 62%) and (25%, 64%) in zero-shot and few-shot settings, respectively. We also evaluate models by providing relevant knowledge statements required to answer the question. We find that the additional knowledge leads to a 7% gain in performance, but the overall performance still remains low. These results make one wonder how much commonsense knowledge about action feasibility is encoded in state-of-the-art models and how well they can reason about it. [1]

1 Introduction

Commonsense reasoning has been a key aspect of AI since its advent in the 1950s. It is closely associated with reasoning about actions and their effects, which is considered a big challenge, especially for deep learning-based AI approaches and language models (LeCun, 2022; Dalvi et al., 2018; Banerjee et al., 2020). While several datasets have been developed to evaluate large-scale language models, in this paper, we propose a dataset focused on reasoning about actions and their effects; specifically, the ability to reason if an action or its effect is feasible.

Figure 1: Illustrating a binary classification (BCQ) and a multiple choice question (MCQ) from FeasibilityQA. The correct answer options (**False** in BCQ and **(45, 37)** in MCQ) are highlighted in **bold**.

Figure 1 illustrates an example of our dataset; given the information "*Sam gave 50 dollars to the shopkeeper to buy a book and the shopkeeper returned some money*", it is not possible to compute the exact price of the book; however, it can be established that the feasible price of the book is *less than* 50 since the shopkeeper returned some money. We often come across such questions in our daily life and find it trivial to reason about them. Therefore, in order to develop NLP systems that can reliably reason about real-world situations, it is important to evaluate their understanding of feasibility.

Recently, many datasets have been created that test different reasoning skills such as pronoun resolution (Sakaguchi et al., 2021; Levesque et al., 2012), commonsense reasoning (Singh et al., 2021; Mihaylov et al., 2018; Banerjee et al., 2021), numerical reasoning (Mishra et al., 2022b; Ravichander et al., 2019; Lin et al., 2020; Zhang et al., 2020; Amini et al., 2019; Mishra et al., 2022e; Creswell et al., 2022; Pal and Baral, 2021), qualitative reasoning (Tafjord et al., 2019b,a), discrete reasoning (Dua et al., 2019), and temporal reasoning (Zhou et al., 2019). However, they do not have an ample number of examples that test understanding of feasibility.

---

[1] Dataset, baseline approaches, and instruction-tuned modeling approaches are freely available at `https://github.com/kevinscaria/feasibilityQA`

Category	Knowledge	Context	Questions
Attribute Comparison	Larger volume holds more amount of water.	Barrett's has two cylindrical shaped bottles. A bottle with a higher volume holds 32 units of water.	Amount of water in other bottle could be 28 units. Amount of water in other bottle could be 33 units. What could be the amount of water in other bottle?
Change with Time	Age increases with time	Today, while filling the application form Edward filled the age field with 16.	Edward could have been 8 years old on his last birthday. Edward could have been 19 years old on his last birthday. What could have been Edward's age last year?
Change with Action	Selling something reduces its quantity	Joshua organized a garage sale yesterday. Joshua sold a total of 273 items at a minimum price of 1 USD each.	Joshua could have made 300 dollars in the garage sale. Joshua could have made 260 dollars in the garage sale. How much money Joshua could have made from the garage sale?
Implicit Numerical Knowledge	4 quarters make 1 dollar	Christopher is accepting quarters for a donation and fails to collect 12 dollars.	He could have collected 35 quarters. He could have collected 52 quarters. How many quarters could Christopher have collected?
Non Numerical	New movies can be watched after the release date.	The latest superhero movie was releasing on 28th February 2022. Ashton wanted to see the movie.	He could have watched the movie on 2nd March 2022. He could have watched the movie on 3rd February 2022. When could Ashton have watched the movie?

Table 1: Illustrative examples of two binary choice questions and one multiple choice question corresponding to a context statement. We also provide the corresponding category and relevant knowledge for the questions.

In this work, we address the above limitation and introduce FeasibilityQA, a dataset consisting of questions that require an understanding of feasibility. This dataset comprises of two types of questions: binary classification (BCQ) and multi-choice multi-correct questions (MCQ). In BCQ, the task is to determine whether the question is feasible or not given a context; in MCQ, the task is to select all feasible answers to the given question. The dataset consists of ∼5K instances covering diverse aspects of feasibility. Table 1 illustrates examples of various categories of feasibility questions.

We conduct comprehensive experiments with GPT-3, GPT-2, and T5 models (Brown et al., 2020; Radford et al., 2019; Raffel et al., 2020) in zero-shot and few-shot settings and show that all of these models struggle to correctly answer feasibility questions. Specifically, on (MCQ, BCQ) questions, GPT-3 achieves an accuracy of just (19%, 62%) and (25%, 64%) in zero-shot and few-shot settings, respectively.

Prior work has found that explicitly providing relevant knowledge helps the model reason better and improves its performance (Chen et al., 2018; Xiong et al., 2019; Banerjee et al., 2019; Varshney et al., 2022a). We explore this aspect of reasoning by explicitly providing relevant knowledge statements and find that it leads to ∼7% improvement in performance. However, the overall performance still remains low. We further investigate GPT-3's ability to reason about feasibility questions by prompting it to generate the reasoning chain. In many cases, we find that GPT-3 successfully generates the correct reasoning chain but still fails to output the correct answer. This analysis further leads to several interesting findings (Section 3).

2 FeasibilityQA

2.1 Dataset Creation

For creating data instances of FeasibilityQA, we first create a context statement that describes a real-life situation. Then, we write two binary classification questions and one multiple choice question conditioned on the context that tests the understanding of feasibility.

Dataset creation and verification process Seven computer science graduate students were involved in creating the dataset. Dataset creation consists of 3 phases. First, in the data creation stage, each student created 700 samples over the period of 3 months. In the next phase, each dataset creator's questions were verified by a different student to ensure fairness during data validation. The 3rd stage of the validation was done when all the questions were compiled and cross-verified. In each verification stage, the dataset creators rejected some samples where the inter-annotator agreement was low. [2]

2.1.1 Context Creation

We create context statements from real-life situations spanning diverse topics such as elementary physics, profit-loss scenarios, temporal comparisons, and quantity comparisons. We divide the contexts into the following five categories:

Attribute comparison: This category consists of questions that test feasibility aspects involving the comparison of attributes of two quantities. **Implicit numerical:** Questions in this category involve fundamental mathematical facts that test the ability to use those facts in real-world situations. **Change with action:** This category tests the abil-

---

[2]All the dataset creators are authors of the paper.

ity to perceive a change in an item or state as an outcome of an action. **Change with time:** Here, questions test the understanding of feasibility related to temporal-based events. **Non Numerical:** This category includes questions where numbers are not explicitly involved in reasoning about feasibility. Table 1 provides examples of these categories. More details about them are in Appendix A.

Motivation behind category selection The motivation behind developing a large language model such as GPT-3 is to mimic human intelligence and come closer to Artificial General Intelligence. We attempt to gauge the performance of models' intelligence by developing simple commonsense reasoning questions. GPT-3 models are few-shot learners but find it hard to do proper numerical reasoning. Earlier datasets like this attempted to analyze numerical reasoning in this aspect. We are also trying to study it in the aspect of feasibility. Previous datasets, such as Numersense (Lin et al., 2020) and MC Taco (Zhou et al., 2019), do not have such a category, and we tried to bridge those gaps.

We think that these five categories are also a good representation of numerical feasibility. We found that questions from those categories had an adequate amount of complexity that the average human could easily figure out. So we expected that large language models should also be able to understand and answer accordingly. We created these categories to compare the models' numerical reasoning ability with and without knowledge. This gives us insights into whether knowledge helps in each aspect. We hope that these comparative studies across these five preliminary categories will inspire more future categories.

Target of our dataset: Our selection of categories in feasibility is inspired by the limitations in existing datasets since it is not possible to cover all the aspects of feasibility exhaustively.

2.1.2 Question Creation

From each context, we create two binary classification and one multiple-choice question. Recall that in our questions, the context may not provide sufficient information to find the exact answer. However, the information is sufficient to test the validity of question/answer options (notice the use of the word '**could**'). In question creation, we ensure that all our contexts and questions describe realistic situations, e.g., we keep a range of numerical entities

Category	Binary Instances	MCQ Instances
Attribute comparison	1696	848
Non numerical	700	350
Implicit numerical	444	222
Change with action	196	98
Change with time	36	18
Total	3072	1536

Table 2: Categorization of FeasibilityQA across different categories .

Correct Answers	#Instances
1	555
2	622
3	285
4	31
None	43

Table 3: Frequency of correct answers for MCQ section.

and units appropriate for their context. Table 1 illustrates examples of our dataset.

BCQ: For each context, we create two binary classification questions where the correct answer is *True* for one and *False* for the other. Evaluating models' consistency in answering two contrasting hypotheses conditioned on the same context provides an additional benefit.

MCQ: For each context, we create a multi-correct multiple-choice question. Here, a context-question pair and the corresponding answer options are given, and the task is to select all feasible options for the question. We ensure that there is also a *None* option, which needs to be selected when all the other options are not feasible. For a question, *one or more options* (including 'None') could be correct.

2.2 Dataset Statistics

Our dataset consists of 1536 contexts and 4608 context-question pairs (3072 BCQ and 1536 MCQ). The category-wise distribution of the dataset is present in Table 2. BCQ dataset is label balanced, i.e., 1536 instances for each of *True* and *False* labels. MCQ dataset has a varying number of correct options. Table 3 shows the number of correct answers in the MCQ category.

	GPT-3				GPT-2				T5			
	BCQ (%)		MCQ (%)		BCQ (%)		MCQ (%)		BCQ (%)		MCQ (%)	
	w/o K	w/ K	w/o K	w/ K	w/o K	w/ K	w/o K	w/ K	w/o K	w/ K	w/o K	w/ K
Zero Shot	62.96	69.11	19.43	25.89	50.00	50.00	0.19	0.45	50.55	50.64	0.13	0.39
One Shot	57.94	64.66	20.94	24.15	50.00	50.00	0.58	1.69	50.61	50.33	0.45	0.58
Few Shot	64.72	68.55	25.94	37.23	50.00	50.00	0.97	0.39	49.81	49.87	0.84	1.10

Table 4: *Exact match accuracy* of three models in BCQ (Binary Classification) and MCQ (multi-choice multi-correct) tasks across three settings. w/o K and w/ K represents without knowledge and with knowledge respectively.

3 Experiments

3.1 Performance Evaluation & Metrics

For **BCQ**, we calculate *exact match accuracy* against the gold label (True or False) for each question. We also use a stricter evaluation metric, *pairwise accuracy*, to better estimate the model's capability of understanding feasibility. Here we only consider a sample to be correct if both BCQ (True and False questions) are correctly answered by the model for one context statement. For **MCQ**, we evaluate *exact match accuracy*, i.e., the model's prediction is considered to be correct if all the correct answer options are predicted. We also compute *recall*, which we define as the number of correct answers predicted by the model out of all the correct answer options. Other results (false positive, false negative, category-wise) are in Appendix B.

Models: We evaluate the performance of **GPT-3** (Text-DaVinci-002, with 256 max tokens, top p of 1, and frequency & presence penalty of 0), **T5-11B**, and **GPT-2 large** models on our dataset.

3.2 Results

Low Performance of All Models: Table 4 shows the accuracy of all three models in zero-shot, one-shot, and few-shot settings. On BCQ, GPT-3 achieves *exact match accuracy* of just 62.9%. GPT-2 and T5 perform even worse and achieve close to the majority baseline (50%). GPT-2 gets an exact 50, indicating that the model does not understand such feasibility reasoning [3]. On MCQ, which is a more difficult task than BCQ, all models, including GPT-3, achieve a very low *strict accuracy* score. This highlights that feasibility questions are challenging for even state-of-the-art models.

Decrease in performance in one-shot setting: In the one-shot setting, the model's prediction is heavily influenced by the label of the example (one) presented to the model. This phenomenon is also observed in several prior zero-shot, and one-shot

	w/o Knowledge	w/ Knowledge
Zero	46.2	49.9
One	63.5	64.5
Few	66.7	70.3

Table 5: *Recall* scores of GPT-3 on MCQ task.

	w/o Knowledge	w/ Knowledge
Zero Shot	42.9	56.8
One Shot	17.9	34.3
Few Shot	39.8	55.8

Table 6: *Pairwise Accuracy* of GPT-3 in BCQ Task.

studies (Zhao et al., 2021). A similar trend is observed in the chain of thought experiments (results described in Table 8).

Providing Knowledge Improves GPT-3's performance by $\sim 7\%$ across all settings. The accuracy particularly increases ($\sim 12\%$) in the MCQ task in the few-shot setting. Although GPT-3 performs better than T5 and GPT-2, it achieves just 68.5% and 37.2% on BCQ and MCQs, respectively.

GPT-3 achieves High Recall Scores on MCQs: In Table 5, we show recall scores of GPT-3 on MCQs. GPT-3 achieves a high score (up to 70%), highlighting that it gives correct responses but fails to give all the correct responses.

Pairwise Evaluation: Recently, instance-level analysis of the evaluation data has received considerable research attention (Zhong et al., 2021; Varshney et al., 2022b; Rodriguez et al., 2021; Mishra et al., 2022a). Motivated by this, we analyze GPT-3's performance on BCQ questions using the stricter *pairwise accuracy* metric. Even though the model performs $\sim 63\%$ using *exact match accuracy*, Table 6 shows that the models' performance is at most $\sim 43\%$ via *pairwise accuracy*, highlighting a performance gap. The accuracy increases ($\sim 13\%$) when knowledge is introduced, and the gap between different settings also narrows down, indicating that the addition of knowledge helps.

---

[3]Please refer Appendix B.2 for details

	BCQ		MCQ	
	w/o K	w/ K	w/o K	w/ K
Attribute Comparison	58.2	62.5	17.7	23.2
Non Numerical	77.2	89.4	23.6	30.9
Implicit Numerical	54.7	50.9	16.7	21.9
Change with Action	66.3	78.2	20.3	26.6
Change with Time	58.3	66.6	17.8	23.4

Table 7: Category wise *Exact Match Accuracy* of GPT-3 on BCQ and MCQ in one-shot setting.

Figure 2: Answers with explanations given by GPT-3 on FeasibilityQA dataset.

	BCQ		MCQ	
	w/o K	w/ K	w/o K	w/ K
Zero Shot	61.3	70.2	20.1	25.1
One Shot	59.7	67.2	19.5	22.8
Few Shot	65.4	69.1	23.4	35.7

Table 8: *Exact Match Accuracy* of GPT-3 on BCQ and MCQ tasks with chain of thought setting.

Category wise results: Table 7 shows the category-wise results in one-shot setting for BCQ and MCQ tasks on GPT-3. Non Numerical category performed the best out of all categories (15% more than other categories). Addition of knowledge improves the performance of all categories by around 5% to 10%. Similar trends are observed across MCQ task as well. We believe that Non Numerical performed the best because these samples were conditioned around factual knowledge and that it could be present in the pretraining corpus of GPT-3. We further analyze this in Appendix B.

Investigating Chain of Thoughts Prompting: Recent work has demonstrated the success of learning from instructions (Wei et al., 2021; Wang et al., 2022; Mishra et al., 2022d,c; Lu et al., 2022; Parmar et al., 2022; Mishra and Nouri, 2022; Luo et al., 2022) and chain of thought (Wei et al., 2022) and

scratchpad prompting (Nye et al., 2021). To test this on FeasibilityQA, we add explanations to one-shot and few-shot examples provided in the context. Table 8 shows marginal improvement. More details are in Appendix C.

A Case Study on Prompting GPT-3 to Provide Explanation: We further investigate the reason behind GPT-3's poor performance on FeasibilityQA by prompting it to provide the reason behind its answer. Specifically, we add *"Explain the reason behind your answer"* in the prompt. Figure 2 illustrates a response from GPT-3. The answer demonstrates that it did not understand the numerical value of Abraham's age. We also provide additional knowledge to assist the model, as shown in Figure 2. Even with knowledge, the model could not understand the feasible age.

4 Conclusion

In this work, we proposed FeasibilityQA, a question-answering dataset that evaluates the understanding of feasibility. We conducted extensive experiments with several state-of-the-art models in zero-shot, one-shot, and few-shot settings and show that these models struggle to answer the feasibility questions correctly. We also experimented by providing additional knowledge (relevant to the question) and showed that it leads to a small gain in performance, but the overall performance still remains low. We further analyzed the performance of models that reveals several interesting findings. Finally, we release our dataset and hope that our work will encourage further research in feasibility reasoning, an important yet underexplored aspect of commonsense reasoning.

Limitations

Like any other commonsense reasoning ability, the concept of feasibility, in general, can be applied in numerous real-world situations. In our dataset, we try to cover a diverse set of such situations that test the understanding of feasibility, but it is in not an exhaustive list. In the future, we will expand the category space by either converting existing numerical datasets into feasibility questions or manually creating new category spaces. Along with the dataset, we release the list of contexts and situations on which the question is based. In the future, this would help expand the dataset to cover other domains and situations. The human evaluation of the dataset could also be an interesting

study, but it can be an expensive. The selection of humans in terms of their educational background and age is also required for unbiased evaluation. A completely random selection of humans is also required for a comprehensive study. Finally, our dataset includes questions in only one language, i.e., English.

Ethical Considerations

The names used in this dataset are selected from the most common English names. In question creation, we ensure that all our contexts and questions describe realistic situations, e.g., we keep a range of numerical entities and units appropriate for their context. No personal information from data creators has been collected during the creation of the dataset.

References

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

Pratyay Banerjee, Chitta Baral, Man Luo, Arindam Mitra, Kuntal Pal, Tran C Son, and Neeraj Varshney. 2020. Can transformers reason about effects of actions? *arXiv preprint arXiv:2012.09938*.

Pratyay Banerjee, Swaroop Mishra, Kuntal Kumar Pal, Arindam Mitra, and Chitta Baral. 2021. Commonsense reasoning with implicit knowledge in natural language. In *3rd Conference on Automated Knowledge Base Construction*.

Pratyay Banerjee, Kuntal Kumar Pal, Arindam Mitra, and Chitta Baral. 2019. Careful selection of knowledge to solve open book question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6120–6129, Florence, Italy. Association for Computational Linguistics.

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

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, and Diana Inkpen. 2018. Natural language inference with external knowledge.

Antonia Creswell, Murray Shanahan, and Irina Higgins. 2022. Selection-inference: Exploiting large language models for interpretable logical reasoning. *arXiv preprint arXiv:2205.09712*.

Bhavana Dalvi, Lifu Huang, Niket Tandon, Wen tau Yih, and Peter Clark. 2018. Tracking state changes in procedural text: a challenge dataset and models for process paragraph comprehension. In *North American Chapter of the Association for Computational Linguistics*.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. 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 2368–2378, Minneapolis, Minnesota. Association for Computational Linguistics.

Yann LeCun. 2022. A path towards autonomous machine intelligence. *preprint posted on openreview*.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The Winograd Schema Challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'12, pages 552–561. AAAI Press, Rome, Italy.

Bill Yuchen Lin, Seyeon Lee, Rahul Khanna, and Xiang Ren. 2020. Birds have four legs?! NumerSense: Probing Numerical Commonsense Knowledge of Pre-Trained Language Models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6862–6868, Online. Association for Computational Linguistics.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and A. Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *ArXiv*, abs/2209.09513.

Man Luo, Sharad Saxena, Swaroop Mishra, Mihir Parmar, and Chitta Baral. 2022. Biotabqa: Instruction learning for biomedical table question answering. *arXiv preprint arXiv:2207.02419*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Conference on Empirical Methods in Natural Language Processing*.

Swaroop Mishra, Anjana Arunkumar, Chris Bryan, and Chitta Baral. 2022a. Hardness of samples need to be quantified for a reliable evaluation system: Exploring potential opportunities with a new task. *ArXiv*, abs/2210.07631.

Swaroop Mishra, Matthew Finlayson, Pan Lu, Leonard Tang, Sean Welleck, Chitta Baral, Tanmay Rajpurohit, Oyvind Tafjord, Ashish Sabharwal, Peter Clark, and A. Kalyan. 2022b. Lila: A unified benchmark for mathematical reasoning. *ArXiv*, abs/2210.17517.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, Yejin Choi, and Hannaneh Hajishirzi. 2022c. Reframing instructional prompts to GPTk's language. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 589–612, Dublin, Ireland. Association for Computational Linguistics.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022d. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487, Dublin, Ireland. Association for Computational Linguistics.

Swaroop Mishra, Arindam Mitra, Neeraj Varshney, Bhavdeep Sachdeva, Peter Clark, Chitta Baral, and Ashwin Kalyan. 2022e. NumGLUE: A suite of fundamental yet challenging mathematical reasoning tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3505–3523, Dublin, Ireland. Association for Computational Linguistics.

Swaroop Mishra and Elnaz Nouri. 2022. Help me think: A simple prompting strategy for non-experts to create customized content with models. *arXiv preprint arXiv:2208.08232*.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. 2021. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*.

Kuntal Kumar Pal and Chitta Baral. 2021. Investigating numeracy learning ability of a text-to-text transfer model. In *Conference on Empirical Methods in Natural Language Processing*.

Mihir Parmar, Swaroop Mishra, Mirali Purohit, Man Luo, M. Hassan Murad, and Chitta Baral. 2022. Inboxbart: Get instructions into biomedical multi-task learning. In *NAACL-HLT*.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

Abhilasha Ravichander, Aakanksha Naik, Carolyn Rose, and Eduard Hovy. 2019. EQUATE: A benchmark evaluation framework for quantitative reasoning in natural language inference. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 349–361, Hong Kong, China. Association for Computational Linguistics.

Pedro Rodriguez, Joe Barrow, Alexander Miserlis Hoyle, John P. Lalor, Robin Jia, and Jordan Boyd-Graber. 2021. Evaluation examples are not equally informative: How should that change NLP leaderboards? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4486–4503, Online. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Shikhar Singh, Nuan Wen, Yu Hou, Pegah Alipoormolabashi, Te-lin Wu, Xuezhe Ma, and Nanyun Peng. 2021. COM2SENSE: A commonsense reasoning benchmark with complementary sentences. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 883–898, Online. Association for Computational Linguistics.

Oyvind Tafjord, Peter Clark, Matt Gardner, Wen-tau Yih, and Ashish Sabharwal. 2019a. Quarel: A dataset and models for answering questions about qualitative relationships. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7063–7071.

Oyvind Tafjord, Matt Gardner, Kevin Lin, and Peter Clark. 2019b. QuaRTz: An open-domain dataset of qualitative relationship questions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5941–5946, Hong Kong, China. Association for Computational Linguistics.

Neeraj Varshney, Man Luo, and Chitta Baral. 2022a. Can open-domain qa reader utilize external knowledge efficiently like humans? *arXiv preprint arXiv:2211.12707*.

Neeraj Varshney, Swaroop Mishra, and Chitta Baral. 2022b. ILDAE: Instance-level difficulty analysis of evaluation data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3412–3425, Dublin, Ireland. Association for Computational Linguistics.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022. Benchmarking generalization via in-context instructions on 1,600+ language tasks. *EMNLP*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *ICLR*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Wenhan Xiong, Mo Yu, Shiyu Chang, Xiaoxiao Guo, and William Yang Wang. 2019. Improving question answering over incomplete KBs with knowledge-aware reader. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4258–4264, Florence, Italy. Association for Computational Linguistics.

Xikun Zhang, Deepak Ramachandran, Ian Tenney, Yanai Elazar, and Dan Roth. 2020. Do language embeddings capture scales? In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4889–4896, Online. Association for Computational Linguistics.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, pages 12697–12706. PMLR.

Ruiqi Zhong, Dhruba Ghosh, Dan Klein, and Jacob Steinhardt. 2021. Are larger pretrained language models uniformly better? comparing performance at the instance level. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3813–3827, Online. Association for Computational Linguistics.

Ben Zhou, Daniel Khashabi, Qiang Ning, and Dan Roth. 2019. "going on a vacation" takes longer than "going for a walk": A study of temporal commonsense understanding. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3363–3369, Hong Kong, China. Association for Computational Linguistics.

Appendix

A   Dataset characteristics

In this section, we describe FeasibilityQA in more detail. Table 1 shows illustrative examples of each category discussed in section 4. Each row of the 4th Column of the table shows three questions that were prepared in response to a context. Table 2 gives the distribution of each category of the dataset. We will explain the motivation behind each category. Please note that the explanations are with respect to examples presented in Table 1.

Attribute Comparison shows the comparative properties between two similar objects. The context from attribute comparison is designed to show that quantities can be measured using words like higher and lower, and the model has to understand the relation between them to answer different questions. In this example, it is not possible that the smaller bottle can have a volume of 33 units since the larger one is 32 units.

Change with Time gives the events that have time as the changing factor. The context is designed to test the model's ability to deduce time-based changes and how certain actions/ events/ quantities can or cannot be done before/ after a certain time. In this case, it is impossible that Edward's age could be 19 on his last birthday as his current age is 16.

Change with Action describes the actions which alter certain quantities/events and test the model's ability to understand that. In this case, it is demonstrated that selling/giving away a certain quantity reduces it. In the example, it is demonstrated that selling all 273 items at least 1 dollar will leave Joshua with at least 273 dollars. Hence the question that he could have 260 dollars is false.

Implicit Numerical Knowledge tests the model's ability to understand numerical entities as facts and how to manipulate them in different situations. In this case, using the knowledge (or without using it) that four quarters make 1 dollar, the model needs to understand how many quarters will be used in 12 dollars, which is 48 quarters. Hence the question tells us that Christopher can have 52 quarters.

Non Numerical category tests the model's understanding of very broad domains. They do not have to be numerical based in all the cases.

The dataset contains diverse topics ranging from physics, mathematics, biology, and numerical reasoning. A total of 422 subcategories are present in the dataset. Table 2 shows the distribution of BCQ and MCQ questions across different categories in the dataset.

B   Other performance results

B.1   Performance Metrics

For the MCQ setting of the dataset, we study true positive, false positive, and false negative rates as the evaluation metrics. Each metric definition is listed below:

False Negative rate is defined as the number of

	w/o Knowledge	w/ Knowledge
Zero	0.17	0.13
One	0.36	0.32
Few	0.33	0.24

Table 9: False Positive rate of GPT-3 on MCQ section

	w/o Knowledge	w/ Knowledge
Zero	0.42	0.42
One	0.21	0.24
Few	0.18	0.20

Table 10: False negative rate of GPT-3 on MCQ section

Zero shot BCQ		
	w/o Knowledge	w/ Knowledge
Attribute Comparison	51.2	55.8
Non Numerical	72.7	85.7
Implicit Numerical	52.9	52.0
Change with Action	60.7	65.3
Change with Time	55.5	55.5
Zero shot MCQ		
	w/o Knowledge	w/ Knowledge
Attribute Comparison	17.9	20.4
Non Numerical	25.5	29.0
Implicit Numerical	18.5	21.1
Change with Action	21.3	24.2
Change with Time	19.4	22.2

Table 11: Category wise Accuracy of GPT-3 on BCQ and MCQ task in zero-shot setting.

incorrect predictions the model gave as correct. For example, if the model gave output as A, B, C, and the predicted result is A, C, then B is missed. The number of false negatives would be 1 (B).

False Positive rate is defined as the number of correct predictions the model gave as incorrect. For example, if the given output is A, B and the predicted result is A, B, C, then the number of false negatives would be 1 (C).

B.2 Results

False positive results shown in Table 9 follow trends similar to accuracy where the performance of one-shot experiments is worse than zero-shot and few-shot. But with the addition of knowledge, the false positive rate decreases.

As shown in Table 10, the False negative rate decreases from zero-shot to few-shot experiments, but contrary to other experiments, it increases with the addition of knowledge in almost all the cases.

Table 11 shows the category-wise results in zero-shot settings for BCQ and MCQ tasks. For the BCQ task, accuracy was lowest in the Attribute compari-

Few shot BCQ		
	w/o Knowledge	w/ Knowledge
Attribute Comparison	64.5	69.5
Non Numerical	85.9	99.4
Implicit Numerical	60.8	56.6
Change with Action	73.7	86.8
Change with Time	64.8	74.1
Few shot MCQ		
	w/o Knowledge	w/ Knowledge
Attribute Comparison	25.2	35.8
Non Numerical	33.5	47.6
Implicit Numerical	23.7	33.7
Change with Action	28.8	40.9
Change with Time	25.3	35.9

Table 12: Category wise Accuracy of GPT-3 on BC and MCQ task in few-shot setting.

son category and highest in Non-Numerical Category. Performance of the Non Numerical category improved significantly in with knowledge setting.

In the MCQ portion of the dataset, the performance gap between Non-Numerical and other categories reduces significantly. It is still the best-performing category for the model, but the Change with Action Category also produced similar results. There was no significant improvement in both Non-Numerical and change with action as observed in the Non-Numerical with the addition of knowledge.

Table 12 shows the category-wise results for BCQ and MCQ tasks in few shot setting. For the BCQ task, accuracy was lowest in the Attribute comparison category and highest in Non-Numerical Category. Performance of the Non Numerical category improved significantly in the knowledge setting with accuracy reaching above 90% for the first time in any of the categories.

In the MCQ portion of the dataset, the performance gap between Non-Numerical and other categories reduces significantly. It is still the best-performing category for the model. There was a significant improvement in Non Numerical and change with action and change with time categories with the addition of knowledge.

Exact 50% accuracy of GPT-2: The input format for all models was as follows: Zero-Shot, Question (Different questions), and Options (True or False). Example(s) preceded the question in the one-shot and few-shot settings. Based on this format, GPT-2 gave the probability of "False" higher in all cases. Since the dataset is label balanced, all the True hypothesis questions were incorrectly predicted, hence giving a 50% accuracy.

Example or Evaluation sample		Text / Context
Example fed in GPT-3 as 1 shot learning example	**Prompt**	Select the best option for the questions based on the Context and explain the reasoning. An Example to show how it works:
	Context	Eric boiled a glass of water to drink the hot water. The level of the water in the glass is 63 units. After boiling was done, she took the water into the glass.
	Question	What could be the level of the water after boiling?
	Options	A: 65, **B: 59**, C: 64, **D: 61**, E: None
	Answer:	Since boiling leads to evaporation, there is a possibility that the volume of water would have decreased. Due to this, the current water level should be less than 63 units. Hence the answer is B:59 and D: 61
Example given to GPT-3 for evaluation	**Context**	A negative number was added to 70.
	Question	What could be the result?
	Options	A: 70, B: 89, C: 87, **D: 68**, E: None
	GPT-3 Answer	The result could be A: 70, since **adding a negative number would not change the value of 70**. The result could also be C: 87, since adding a negative number would increase the value of 70
Example given to GPT-3 for evaluation	**Context**	Last year 758 people joined the company and this year the number of people who joined was more than double the last time
	Question	How many people could have joined this year?
	Options	A: 1440, **B: 1600**, C: 1460, D: 1370, E: None
	GPT-3 Answer:	Since the number of people who joined this year was **more than double the last time**, the number of people who joined this year could be up to 1460. Therefore, the answer is C: 1460.

Table 13: Illustrating chain of thought approach on some examples of feasibilityQA dataset in 1 shot setting without providing knowledge. 1st set of rows demonstrate the example fed into GPT-3 for 1 shot learning. 2nd and 3rd set of rows show GPT-3's response to Context, Question and Options asked.

Example or Evaluation sample		Text / Context
Example fed in GPT-3 as 1 shot learning example	**Prompt**	Select the best option for the questions based on the context, knowledge and explain the reasoning. An Example to show how it works:
	Context	Eric boiled a glass of water to drink the hot water. The level of the water in the glass is 63 units. After boiling was done, she took the water into the glass.
	Knowledge	Water will convert into water vapour on boiling.
	Question	What could be the level of the water after boiling?
	Options	A: 65, B: 59, C: 64, D: 61
	Answer	Since boiling leads to evaporation, there is a possibility that the volume of water would have decreased. Due to this, the current water level should be less than 63 units. Hence the answer is B:59 and D: 61
Example given to GPT-3 for evaluation	**Context**	Two boys competed in a race. The loser finished the race in 24 minutes.
	Knowledge	Loser will take more time to finish the race.
	Question	How many minutes the other boy could have taken to finish the race?
	Options	A: 32, B: 25, C: 15, D: 22, E: None
	GPT-3 Answer	The other boy **could have taken less time to finish the race**. Hence, the answer is B: 25.

Table 14: Illustrating chain of thought approach on some examples of feasibilityQA dataset in 1 shot setting with providing knowledge. 1st set of rows demonstrate the example fed into GPT-3 for 1 shot learning. 2nd row shows GPT-3's response to Context, Question and Options asked.

C Case study: Chain of Thought Reasoning Approach

Table 13 and 14 show the unsuccessful attempts in the chain of thought reasoning approach. Table 13 shows the setting where the 1st example is fed into the model as an example of how to reason out the answer. The reason and answer were clearly mentioned that told that evaporation leads to a decrease in water level and hence water level should decrease. This led to a decrease in water level; hence, the correct answers were quantities less than 63; **59 and 61**.

The 2nd and 3rd sets of rows show the Context, question, and options supplied to GPT-3, and we get responses in GPT-3 Answer row. The logic given for the addition of a number is wrong.

Adding a negative number should decrease the value, and hence rest of the answer will be wrong. In the 3rd row GPT-3's response, the logic used to calculate the answer is correct, but it was unable to calculate that 1600 was double 758. Both parts are highlighted in the table.

The situation did not improve much when knowledge was supplied with other rows, as shown in Table 14. The model was able to interpret the logic correctly but could not associate that logic with numerical quantities.