LogicPrpBank: A Corpus for Logical Implication and Equivalence

Zhexiong Liu[1*], **Jing Zhang** [2*], **Jiaying Lu** [2], **Wenjing Ma**[2], **Joyce C Ho**[2]
[1] University of Pittsburgh [2] Emory University
zhexiong@cs.pitt.edu {jing.zhang2, jiaying.lu, wenjing.ma, joyce.c.ho}@emory.edu

Abstract

Logic reasoning has been critically needed in problem-solving and decision-making. Although Language Models (LMs) have demonstrated capabilities of handling multiple reasoning tasks (e.g., commonsense reasoning), their ability to reason complex mathematical problems, specifically propositional logic, remains largely underexplored. This lack of exploration can be attributed to the limited availability of annotated corpora. Here, we present a well-labeled propositional logic corpus, LogicPrpBank, containing 7093 Propositional Logic Statements (PLSs) across six mathematical subjects, to study a brand-new task of reasoning logical *implication* and *equivalence*. We benchmark LogicPrpBank with widely-used LMs to show that our corpus offers a useful resource for this challenging task and there is ample room for model improvement.

Keywords: Propositional Logic, Logical Reasoning, Language Models, Few-shot Learning

1. Introduction

Propositional logic deals with propositions (i.e., statements that can be true or false) and logical relationships between propositions. It has been used to solve many scientific problems (e.g., computer logic gates, distributed computing, and game strategies) (Pietarinen, 2001) and facilitate educational applications such as Intelligent Tutoring Systems (ITSs) (Galafassi et al., 2020; Mandal and Naskar, 2021). However, reasoning with propositional logic is different from reasoning in a Natural Language Processing (NLP) task (e.g., causal inference, commonsense reasoning, etc.) as propositional logic is a formal language formed with a set of symbols and rules that are distinct from those in natural languages (Traylor et al., 2021). Table 1 shows a truth table of logical *implication* and *equivalence* using a propositional theory for reasoning. As an illustration, the entailment of Propositional Logic Statement (PLS) P → Q[1], *"If the sum of the interior angles of a triangle is greater than 180 degrees, then a square has five sides"*, is a true statement given both P and Q are false, but these statements (P → Q, P, Q) are all incorrect from the commonsense perspective.

Recently, Language Models (LMs) have demonstrated strong abilities to solve mathematical problems, e.g., approximating solutions to Partial Differential Equations (Li et al., 2022), solving simple math word problems (Patel et al., 2021), and reasoning arithmetic and logical problems (Wang et al., 2021). It has been proven that increasing the scale of LMs (e.g., the size of model parameters) can lead to better performance and sample efficiency in many NLP tasks (Devlin et al., 2019; Brown et al., 2020; Kasneci et al., 2023). However, this claim is questionable in the propositional logic field because LMs are pre-trained with corpora that incorporate rationale in natural languages while reasoning with propositional logic requires understating rationale defined in formal languages. While Traylor et al. (2021) study a simple set of propositional logic (e.g., *and*, *or*, *negation*) by investigating under which conditions LMs can successfully emulate the meaning of formal languages, this only reveals a small portion of propositional logic problems, but more

[*] These authors contributed equally to this work.

[1] Note that statements P and Q do not need to be semantically related in propositional logic.

Table 1: The truth table of logical *implication* (P → Q) and *equivalence* (P ↔ Q). Note that the logical *implication* is always true given statement P is false, regardless of the truth value of statement Q, which is counterintuitive in commonsense.

P	Q	$P \rightarrow Q$	$P \leftrightarrow Q$
T	T	T	T
T	F	F	F
F	T	T	F
F	F	T	T

complex ones (e.g., *implication* and *equivalence*) are underexplored. To bridge the gap, we present LOGICPRPBANK that contains 7093 PLSs in six mathematical subjects (*algebra, arithmetic, calculus, geometry, number theory (numbers),* and *statistics*) to investigate the LMs' capabilities of reasoning complex propositional logic. This endeavor is beneficial to mathematical ITSs.

There have been available corpora for evaluating LMs' abilities to understand and reason. Bowman et al. (2015) present the SNLI corpus that focuses on single-step inferences (e.g., entailment, contradiction, irrelevance) between two pieces of text. SNLI is unable to explain reasoning chains, although an extension has been later implemented in Camburu et al. (2018). Neves Ribeiro et al. (2022) introduce ENTAILMENTBANK corpus that investigates the entailment relations of natural language text. However, these datasets focus on propositional logic inference (e.g., entailing conclusions from premises) but not the fundamental correctness of PLSs in mathematical subjects. Recently, Ontanon et al. (2022) collected a propositional logic corpus LOGICINFERENCE, which validates the subset of first-order propositional logics using sequence-to-sequence LMs. In comparison to LOGICINFERENCE which focuses on reasoning between premises and conclusions, LOGICPRPBANK has two major differences. First, we focus on mathematical PLSs which can be used for building educational applications (e.g., ITSs). Second, instead of dealing with logical inference (e.g., inference chains), LOGICPRPBANK investigates the correctness of PLSs (e.g., the truth values of PLSs).

With LOGICPRPBANK, our work investigates two research questions: **RQ1** Are LMs capable of reasoning complex propositional logic (e.g., *implicating* and *equivalence*) in real mathematical subjects? **RQ2** Are large-scale LMs better than small-scale LMs in reasoning propositional logic? We benchmark LOGICPRPBANK and make the following contributions:

- We leverage the state-of-the-art ChatGPT to generate real atomic PLSs in six mathematical subjects and then develop a proposition composer to compose atomic to compound PLSs.
- We investigate LM's capability of reasoning complex *implication* and *equivalence* PLSs which is different from reasoning in existing NLP tasks.
- We conduct experiments on LogicPrpBank with various scales of LMs to study the pros and cons of LMs in reasoning with propositional logic.

2. Corpus

We use ChatGPT API[2] to generate atomic PLSs. In particular, we develop a data collection prompt to collect True and False PLSs: `please list [X] [Y] atomic statements in [Z],`

[2] https://openai.com/blog/chatgpt

Table 2: The examples of the proposed LOGICPRPBANK corpus.

IDs	Types	Subjects	Propositional Logic Statements	Truth Values
1	Atomic	Arithmetic	The median of 3, 4, 5, 6, 7 is 6.	F
2	Atomic	Geometry	The distance between two parallel lines is the same at all points.	T
3	Atomic	Numbers	The sum of the first n odd integers is n(n+1).	F
4	Implication	Geometry	The distance between two parallel lines is the same at all points is necessary and sufficient for the area of a circle is always pi * r * r, where r is the diameter.	F
5	Equivalence	Calculus	The derivative of log(x) with respect to x is equal to 1/x is equivalent to the implicit function theorem only applies to functions of two variables.	T

where `[X]` is the number of PLSs (e.g., `X`=20), `[Y]` is chosen from *True* or *False*, and `[Z]` is chosen from one of the subjects (*algebra, arithmetic, calculus, geometry, number theory,* and *statistics*). We run the same prompts multiple times until having substantial True and False PLSs for each subject.

We use ChatGPT as a corpus source rather than open sources (e.g., online articles) or human annotations for two reasons. First, ChatGPT is trained with vast amounts of data from the internet written by humans, which covers propositional logic lectures across educational and tutoring webpages, thus it is able to generate a large number of high-quality PLSs. Also, using ChatGPT to generate a corpus is a new exploration of LMs' applications in corpus construction which minimizes labor costs associated with collection and reduces annotation costs. Second, understanding mathematical PLSs necessitates annotators acquire mathematical knowledge at the college or even higher education levels, thus it is not feasible for annotators to create a large number of PLSs from scratch. Therefore, we conduct a pilot exploration of corpus construction using ChatGPT. Note that a ChatGPT-generated PLS contains two-dimensional information: one is the statement itself; another is the truth value of the statement. To validate the correctness (true/false) of ChatGPT-generated PLSs, we employ qualified human annotators who pass a qualification test to check the ChatGPT-generated PLSs. Annotators are asked to check whether or not a ChatGPT-generated PLS is matched to its truth values by using annotators' expert knowledge, checking online resources, or referencing textbooks. Each ChatGPT-generated PLS is checked by one annotator. We observe that the ChatGPT-generated PLSs have a 17.4% error rate, where a ChatGPT-generated false PLS is proved to be True; and vice versa. We then ask annotators to manually correct the wrong statements by revising the PLSs to match their correct truth values. We randomly sample 10% atomic PLSs from each subject and ask two annotators to annotate their truth values without seeing ChatGPT-generated truth values. The Cohen's kappa between the two annotators is 0.77.

To generate *implication* and *equivalence* PLSs, we develop a template-based proposition composer with curated templates to automatically compose two atomic PLSs into one compound PLS. An *implication* composer uses a set of templates: (1) `if [P] then [Q]`; (2) `[P] implies [Q]`; (3) `[P], therefore, [Q]`. An *equivalence* composer uses a set of templates: (1) `[P] if and only if [Q]`; (2) `[P] is necessary and sufficient for [Q]`; (3) `[P] is equivalent to [Q]`. Here `[P]` and `[Q]` denote two different atomic PLSs. Accordingly, the labels (i.e., truth values) of compound PLSs are inferred from their truth table (see Table 1, where *implication* is the column of $P \rightarrow Q$ and *equivalence* is the column of $P \leftrightarrow Q$). We collect 1277 atomic PLSs that cover axioms, theorems, and practice problems. We randomly sample one P and one Q from the same subject to generate compound PLSs ($P \rightarrow Q$ and $P \leftrightarrow Q$) by running the proposition composers. After several rounds of the process, 5816 compound PLSs are generated. Table 2 shows examples from the LOGICPRPBANK corpus, where a compound PLS is composed

Table 3: The statistics of LOGICPRPBANK corpus across subjects, atom, *implication*, and *equivalence* PLSs. The # before/after slash is the # of True and False PLSs, respectively.

Types	Algebra	Arithmetic	Calculus	Geometry	Numbers	Statistics	Total
Atomic	115 / 132	117 / 117	101 / 110	115 / 117	43 / 60	122 / 128	613 / 664
Implication	466 / 144	410 / 138	338 / 107	405 / 133	83 / 23	465 / 160	2167 / 705
Equvilance	338 / 287	295 / 266	236 / 220	299 / 253	53 / 56	363 / 278	1584 / 1360
Total	919 / 563	822 / 521	675 / 437	819 / 503	179 / 139	950 / 566	4364 / 2729

of two atomic PLSs (e.g., the *implication* PLS in Row#4 uses the atomic PLS in Row#2). Table 3 shows the statistics of atomic, *implication*, and *equivalence* PLSs regarding their truth values (true/false) across six mathematical subjects. Note that *implication* has more true PLSs than false PLSs because P→Q is always true given P is false, regardless of the value of Q (see Table 1). The true/false ratio in atom and *equivalence* is near one.

3. Experiments and Analysis

In this section, we introduce the benchmark experiments on LOGICPRPBANK corpus for PLS correctness (true/false) classification. We use small-scale LMs, e.g., DistilRoBERTa (Sanh et al., 2019), RoBERTa-base (Liu et al., 2019), BERT-base and BERT-large (Devlin et al., 2019), medium-scale LMs, e.g., GPT2-medium (Radford et al., 2019), and BLOOM-560m (Scao et al., 2022), and large-scale Language Models (LLMs), e.g., Llama2-7B (Touvron et al., 2023), to reason atomic, *implication*, and *equivalence* PLSs, respectively. We finetune small- and medium-scale LMs with parameter sizes ranging from dozens million to half a billion (given limited computing resources), and perform few-shot learning on LLMs. In particular, we evaluate the performance of Llama2-7B on the test set with zero-shot, 1-shot, 3-shot, 5-shot, and 10-shot learning. In the zero-shot scenario, we predict the test set results without using any training examples. In the other few-shot experiments, we retrieve top-n examples from the training set that are most similar to the test example as context. The similarity is determined by cosine function between sentence embedding (Reimers and Gurevych, 2019). We split the corpus into the train (70%), validation (10%), and test (20%) sets. We use the training set to train small- and medium-scale LMs and use the validation set to tune their parameters. In the training, we optimize the model using Adam optimizer with $\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-8}$. The learning rate is 10^{-5} and the batch size is 32. We train 20 epochs and select the best model on the validation. We conduct three-seed runs and report average macro F1 scores on the test set for both finetuning and zero-shot/few-shot learning. We implement LMs with PyTorch (Paszke et al., 2019) and initial model weights from HuggingFace [3]. Due to space limitation, we introduce prompt details in our code repository[4]. The experiments are running on a GeForce RTX 3090 GPU.

Table 4 shows the results of zero-shot/few-shot and finetuned LMs for identifying the correctness of PLSs in six mathematical subjects. We have observed that small-scale LMs perform excellently in *calculus*, *geometry*, and *statistics* but are dramatically poor in *arithmetic* and *number theory*. This might be because *arithmetic* and *number theory* have many number-related propositions (see Table 2) that are known deficiencies of LMs. These observations answer **RQ1** that LMs are able to reason complex propositional logic only on specific mathematical subjects. Figure 1 shows

[3] https://huggingface.co/

[4] https://github.com/JZCS2018/AI4ED2024.git

Table 4: The F1 scores of zeroshot/fewshot and finetuned LMs on LOGICPRPBANK across six subjects. The top rows are LLMs, the middle rows are medium-scale LMs, and the bottom rows are small-scale LMs. The highest F1 scores are in bold.

Models	Sizes	Algebra	Arithmetic	Calculus	Geometry	Numbers	Statistics	Overall
Llama2-zeroshot	7B	39.07	38.37	34.68	38.53	36.17	40.11	38.28
Llama2-1shot	7B	41.19	38.37	44.43	42.17	40.26	43.89	42.09
Llama2-3shot	7B	45.70	42.29	43.94	45.10	46.67	46.92	45.08
Llama2-5shot	7B	53.30	39.09	54.29	47.69	38.44	48.61	46.94
Llama2-10shot	7B	43.72	39.88	43.29	45.52	35.48	45.39	43.59
BLOOM-560m	560M	39.80	37.96	38.23	39.82	36.23	39.19	38.98
GPT-2-medium	345M	39.08	38.52	37.82	39.34	36.74	38.67	38.69
BERT-large	340M	77.83	49.28	96.40	92.70	52.85	97.03	81.03
RoBERTa-base	125M	83.57	51.71	96.80	95.53	44.22	96.15	83.36
BERT-base	110M	**92.00**	**56.71**	**98.94**	**98.19**	**66.93**	**99.30**	**87.65**
DistilRoBERTa	82M	91.54	55.78	98.77	97.59	56.67	98.87	87.27

the results of finetuned small- and medium-scale LMs and pre-trained LLMs on atomic, *equivalence*, and *implication*. We observe that *implication* is generally better than the other two. BERT and RoBERTa (LMs) have good performance but medium-scale LMs (BLOOM-560m and GPT-2-medium) and LLMs have poor performance. These observations answer **RQ2** that small-scale LMs are able to reason complex propositional logic but Llama2 fails, which suggests that increasing the size of LMs results in performance degradation (see BERT-base v.s. BLOOM-560m). Although large-scale LMs are supposed to have better performance and sample efficiency in many downstream NLP tasks, they do not hold true in reasoning with propositional logic. We argue that propositional logic is a formal language that uses different logic theories from those in natural languages (e.g., commonsense knowledge). Therefore, the medium-scale LMs might not learn propositional logic well given the limited size of the corpus for training. The LLMs are slightly better than the medium-scale LMs, which suggests that LLMs may have potential in learning propositional logics, even in a few-shot learning scenario. Moreover, we observe that the 5-shot learning yields the most favorable overall performance, whereas there is an unexpected decline in performance with the 10-shot configuration. This implies that it is possible that more examples provided to LLMs would introduce more noise. In conclusion, the constructed LOGICPRPBANK is helpful for training small-scale LMs to learn complex propositional logic reasoning in most subjects.

4. Related Work

Previous research has focused on addressing mathematical problems within the field of NLP (Wang et al., 2017; Saxton et al., 2019; Dua et al., 2019). These studies utilize a question-answering framework to tackle these mathematical problems. The Math23L corpus (Wang et al., 2017) consists of basic English contextual information, equations, and corresponding answers, primarily involving arithmetic problems. DeepMind's research (Saxton et al., 2019) investigates reasoning processes in algebra. The DROP corpus (Dua et al., 2019) is a reading comprehension corpus that includes various types of mathematical tasks, such as subtraction and selection. Notably, all answers to its questions can be directly or indirectly inferred from the provided passages. These questions bear similarities to those found in Math23L, and the corpus is sourced from elementary school math

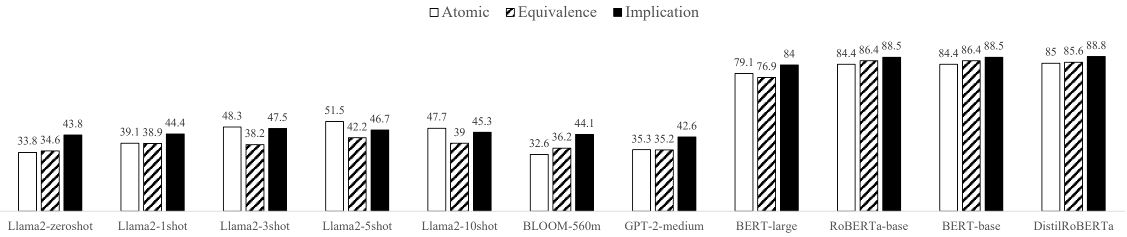

Figure 1: LM performance on LOGICPRPBANK across atom, *implication*, and *equivalence* PLS.

word problems. Based on their limitations, we propose LOGICPRPBANK corpus to address PLSs in six subjects, which poses a brand-new task of reasoning propositional logic.

5. Conclusion

Reasoning propositional logic (e.g., *implication* and *equivalence*) differs from reasoning in NLP tasks (e.g., commonsense reasoning), since the former involves adhering to a set of rules expressed in formal languages, while the latter requires understanding about the real world and the common knowledge conveyed in natural language. To date, there are limited corpora and studies focusing on propositional logic, especially in mathematical subjects. To bridge the gap, we present LOGICPRPBANK, a corpus containing 7093 atom, *implication*, and *equivalence* proposition statements, designed to facilitate LMs to reason complex mathematical propositional logic. The experiments indicate that LLMs (e.g., Llama2-7B) and medium-LMs (e.g., BLOOM-560m) perform worse than the lighter and faster LMs (e.g., BERT and RoBERTa), and LMs struggle with reasoning *arithmetic* and *number theory* but promising in *calculus*, *geometry*, and *statistics*. In future work, we plan to extend the LOGICPRPBANK corpus to encompass a wider range of subjects, including physics and chemistry, in order to support the development of interdisciplinary ITSs.

6. Limitations

Our corpus was collected from ChatGPT and then verified and/or annotated by a qualified annotator, thus there could be annotating errors that would influence the accuracy of the experiments. Using ChatGPT to generate a corpus is a brand-new design, which brings challenges to verify and validate the quality of the data, e.g., the inter-agreement rate used in a traditional data annotation pipeline. But our designed verify-then-correct process for data collecting is proven to save time and labor. Also, ChatGPT is not free for the whole community so it would not be available to researchers from specific areas, thus using ChatGPT to generate or annotate data needs extra ethical considerations. In addition, our corpus is relatively small, which makes it difficult to train or even finetune LLMs. Although zeroshot/fewshot learning with Llama2 shows promising results but still not better than trained/finetuned small-scale LMs, which suggests that LLMs are not always the best options while solving specific tasks that have small annotated data. Furthermore, we only focus on propositional logic in mathematical fields, however, logical reasoning is not limited to math subjects but to many real-world scenarios that we have not covered. Moreover, we only benchmark the corpus on a small number of LMs due to computational resource limitations. And, the error analysis regarding the performance of LMs on our proposed corpus is not extensively studied because most LMs are difficult to visualize and/or explain their reasoning steps (e.g., Llama2, CPT-2, BLOOM, etc.). Therefore, we have limited discussions about error analysis in this work.

Acknowledgements. This work was supported by the NSF award IIS-2145411.

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL https://aclanthology.org/D15-1075.

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. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. e-snli: Natural language inference with natural language explanations. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 9560–9572, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/4c7a167bb329bd92580a99ce422d6fa6-Abstract.html.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 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, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1246. URL https://aclanthology.org/N19-1246.

Cristiano Galafassi, Fabiane FP Galafassi, Eliseo B Reategui, and Rosa M Vicari. Evologic: Intelligent tutoring system to teach logic. In *Intelligent Systems: 9th Brazilian Conference, BRACIS 2020, Rio Grande, Brazil, October 20–23, 2020, Proceedings, Part I 9*, pages 110–121. Springer, 2020.

Enkelejda Kasneci, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, et al. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103:102274, 2023.

Zijie Li, Kazem Meidani, and Amir Barati Farimani. Transformer for partial differential equations' operator learning. *ArXiv preprint*, abs/2205.13671, 2022. URL https://arxiv.org/abs/2205.13671.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv preprint*, abs/1907.11692, 2019. URL https://arxiv.org/abs/1907.11692.

Sourav Mandal and Sudip Kumar Naskar. Classifying and solving arithmetic math word problems—an intelligent math solver. *IEEE Transactions on Learning Technologies*, 14(1):28–41, 2021.

Danilo Neves Ribeiro, Shen Wang, Xiaofei Ma, Rui Dong, Xiaokai Wei, Henghui Zhu, Xinchi Chen, Peng Xu, Zhiheng Huang, Andrew Arnold, and Dan Roth. Entailment tree explanations via iterative retrieval-generation reasoner. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 465–475, Seattle, United States, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.35. URL https://aclanthology.org/2022.findings-naacl.35.

Santiago Ontanon, Joshua Ainslie, Vaclav Cvicek, and Zachary Fisher. Logicinference: A new dataset for teaching logical inference to seq2seq models. *ArXiv preprint*, abs/2203.15099, 2022. URL https://arxiv.org/abs/2203.15099.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are NLP models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.168. URL https://aclanthology.org/2021.naacl-main.168.

Ahti-Veikko Pietarinen. Propositional logic of imperfect information: foundations and applications. *Notre Dame Journal of Formal Logic*, 42(4):193–210, 2001.

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

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv preprint*, abs/1910.01108, 2019. URL https://arxiv.org/abs/1910.01108.

David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. Analysing mathematical reasoning abilities of neural models. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=H1gR5iR5FX.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. Bloom: A 176b-parameter open-access multilingual language model. *ArXiv preprint*, abs/2211.05100, 2022. URL https://arxiv.org/abs/2211.05100.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *ArXiv preprint*, abs/2307.09288, 2023. URL https://arxiv.org/abs/2307.09288.

Aaron Traylor, Roman Feiman, and Ellie Pavlick. AND does not mean OR: Using formal languages to study language models' representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 158–167, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-short.21. URL https://aclanthology.org/2021.acl-short.21.

Cunxiang Wang, Boyuan Zheng, Yuchen Niu, and Yue Zhang. Exploring generalization ability of pretrained language models on arithmetic and logical reasoning. In *Natural Language Processing and Chinese Computing: 10th CCF International Conference, NLPCC 2021, Qingdao, China, October 13–17, 2021, Proceedings, Part I 10*, pages 758–769. Springer, 2021.

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