An Attempt to Improve the Baseline Performance on TRIP

Zhixuan Chen
University of Michigan
shczx@umich.edu

Yichen Zhong
University of Michigan
yichenzh@umich.edu

Abstract

The Tiered Reasoning for Intuitive Physics (TRIP) dataset (Storks et al., 2021) is a benchmark focusing on physical commonsense reasoning. The baseline performance on this benchmark reveals that while language models (LMs) can excel in high-level tasks, they struggle with more nuanced tasks such as physical state classification and conflicting sentence detection. This gap implies a challenge for LMs in executing tasks which require robust reasoning. Our project seeks to enhance the baseline performance on the TRIP benchmark. Key steps in our approach include reproducing the TRIP paper's results, utilizing transfer learning from another commonsense reasoning dataset ART, and fine-tuning an additional pretrained model DistilBERT. We fianlly find that applying transfer learning with ART did not improve TRIP baseline performance as we expected. Besides, the fact that smaller models we use in our approaches achieve performance close to baseline indicates potential issues in the original reasoning system structure.

1 Introduction

In the current era, large-scale, pre-trained language models (LMs) have shown great performance in a wide range of NLP tasks such as named entity recognition, question answering, sentiment analysis, etc. While large LMs are good at giving correct predictions for high-level end tasks, these models' intermediate reasoning process is unknown and questionable. Niven and Kao (2019) found that the excellent performance of LMs is always attributed to spurious statistical cues in the dataset instead of reasoning between contexts.

To probe into how well the LMs can demonstrate proper reasoning process, Storks et al. (2021) introduced Tiered Reasoning for Intuitive Physics (TRIP), a benchmark targeting physical commonsense reasoning. In the TRIP paper, although baseline models including BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019) and DeBERTa (He et al., 2021) achieved high accuracy in the high-level task, i.e., determining which story in a story pair is more plausible, it turns out that these models perform poorly in low-level tasks like physical state classification and conflicting sentence detection. This result means that it is challenging for the LMs to conduct tasks through proper reasoning.

Our project try to improve the baseline performance on TRIP by 1) applying transfer learning from another commonsense reasoning dataset called ART and 2) fine-tuning an additional pretrained model DistilBERT directly on TRIP. The final results show that our approaches do not bring significant improvements to the baseline. We attribute the ineffectiveness of transfer learning to the low accuracy on ART and the difference between ART and TRIP. We also speculate that the non-ideal baseline performance may primarily result from the original reasoning system structure. To truly boost the baseline performance, the architecture used for fine-tuning models should be improved.

2 Related Work

2.1 The TRIP Dataset

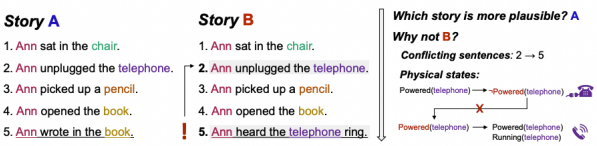

Figure 1: Example story pair from TRIP dataset (Storks et al., 2021)

2.1.1 Dataset Description

Storks et al. (2021) introduced the novel dataset TRIP to advance research in physical commonsense reasoning. The TRIP dataset comprises narratives authored by individuals, each narrative char-

acterized by explicit physical actions. These narratives are organized in pairs, with each pair differentiated by a singular variance—one narrative contains a sentence that renders it implausible. This design facilitates the study of implausibility within the context of physical actions. An example from Storks et al are shown in the Figure 1. Line 5 is the main difference in these two stories, which makes story B implausible. We can use line 2 as a breakpoint, as the telephone can't ring after the telephone is unplugged, there is a conflict in physical states. Hence the story A is more plausible. The TRIP dataset was developed by sourcing plausible stories from Amazon Mechanical Turk, transforming them into implausible ones through sentence replacement by separate workers, followed by a rigorous quality check to eliminate incoherent narratives and correct errors.

The TRIP dataset stands out due to its controlled data curation and emphasis on objectivity in physical commonsense. Recognizing that commonsense can vary based on cultural and regional factors, the dataset was designed to reduce ambiguity and subjectivity by focusing on concrete actions in familiar settings, such as typical household environments. The narratives were confined to simple, declarative sentence structures, emphasizing clarity and directness to minimize linguistic complexity and enhance the focus on reasoning abilities. In addition to the narrative pairs, the TRIP dataset includes multi-tier annotations, providing labels for the physical states involved, identifying conflicting sentences, and justifying the implausibility within each story. This rich annotation process, along with the stringent criteria for narrative construction and the requirement for comprehensive reasoning over extended contexts, makes TRIP an advanced tool for evaluating and advancing AI's understanding of physical commonsense reasoning.

2.1.2 Baseline for TRIP

Baseline System Structure Figure 2 displays a high-level view of the baseline system for TRIP. It individually embeds each sentence-entity pair in each story, classifies physical precondition and effect states, then identifies conflicting sentences from these. Given a pair of stories, it aggregates conflict predictions for each story to decide which is more plausible.

Our project is mainly based on improving the baseline performance of the TRIP dataset proposed by Storks et al. (2021), we need to understand the detail of its baseline structure. There are four modules in this system, each of them is implemented with neural network architecture.(Storks et al. (2021)).

The first module is the **Contextual Embedding**. This module takes as input a sentence and the name of an entity from a story, following an entity-first input formulation (Gupta and Durrett, 2019), and ouputs a dense, contextualized numerical representation.

The second module is called **Precondition and Effect Classifiers**. This module has one precondition classifier and one effect classifier for each of the 20 physical attributes. Altogether, the predictions from these classifiers label physical states of each entity in each sentence of the story (Storks et al., 2021).

The third module is the **Conflict Detector**. This module is going to predict whether the conflicts exist in the entity's physical state, and find a pair of sentences that might cause it. It is implemented using another transformer and additional feedforward classification, and predict the value based on the probability that each sentence conflicting with another sentence in the story. (Storks et al. (2021))

The fourth module is the **Story Choice Prediction**. This module is to output which story is classified as plausible story. It is implemented based on summing negative output and apply softmax from the previous module output.(Storks et al. (2021))

Evaluation Metrics Storks et al. (2021) mentions four evaluation metrics: Accuracy, Consistency, and Verifiability. Those metrics are used to measure the machine's ability to predict reasoning task. Specifically, Accuracy is the proportion of plausible stories are correctly identified. Consistency is the proportion of both plausble stories and conflicting pairs of stories are correctly identified. Verifiability is the proportion of stories that fulfill consistency and identify the underlying changed physical states as well. It is worth mentioned that a if we denote accuracy is a, consistency is b, and verifiability is c, a reliable system model can be demonstrated by a $\approx b \approx c$.

Loss Function There are four loss functions that used during the training: L_p for precondition classification, $\mathcal{L}_p$ for precondition classification, $\mathcal{L}_f$ for effect classification, $\mathcal{L}_c$ for conflicting sentence detection, and $\mathcal{L}_s$ for story choice classification (Storks et al. (2021)).

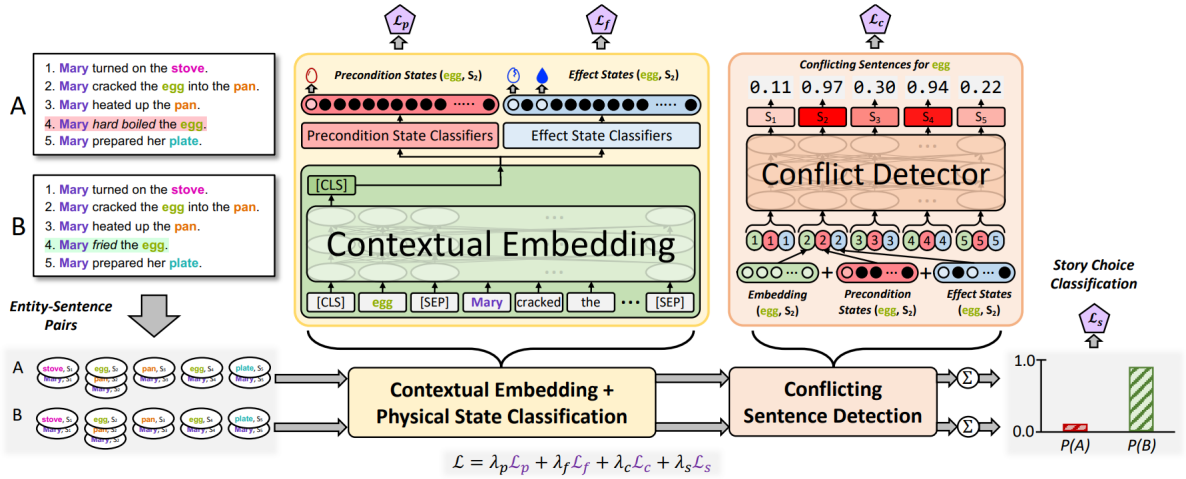

Figure 2: Tiered Reasoning System Structure (Storks et al., 2021)

The baseline TRIP model uses three models to do evaluation: BERT, ROBERTA, and DEBERTA. It also has four types of loss function: All losses, Omit story choice loss $\mathcal{L}_s$, Omit conflict detection loss $\mathcal{L}_c$, and Omit state classification losses $\mathcal{L}_p$ and $\mathcal{L}_f$. The baseline model enumerate all 3*4=12 possible combination cases and using Accuracy, Consistency, and Verifiability metrics to verify. The details are in Figure 3.

Model	Accuracy (%)	Consistency (%)	Verifiability (%)
random	47.8	11.3	0.0
All Losses			
BERT	**78.3**	2.8	0.0
RoBERTa	75.2	6.8	0.9
DeBERTa	74.8	2.2	0.0
Omit Story Choice Loss $\mathcal{L}_s$			
BERT	73.9	**28.0**	9.0
RoBERTa	73.6	22.4	**10.6**
DeBERTa	75.8	24.8	7.5
Omit Conflict Detection Loss $\mathcal{L}_c$			
BERT	50.9	0.0	0.0
RoBERTa	49.7	0.0	0.0
DeBERTa	52.2	0.0	0.0
Omit State Classification Losses $\mathcal{L}_p$ and $\mathcal{L}_f$			
BERT	75.2	17.4	0.0
RoBERTa	71.4	2.5	0.0
DeBERTa	72.4	9.6	0.0

Figure 3: The evaluation result of the originial TRIP paper Storks et al. (2021).

2.2 Recent Progress

Since the release of the TRIP benchmark, a few state-of-the-art language understanding models have been introduced and evaluated upon the TRIP dataset. These models do not only predict the plausibility of a story accurately, but they also show a relatively strong reasoning ability.

Ma et al. (2022) proposed Coalescing Global and Local Information (CGLI), a model that features a novel procedural understanding method. CGLI represents a story or a paragraph with a combination of embeddings from the embedding layer of a LM and timestep embeddings from a *Timestep* embedding layer. In other words, a paragraph will have different embedding for each timestep. In CGLI, a sentence in a paragraph will be assigned with a unique timestep ID where {0=padding, 1=past, 2=current, 3=future}. For timestep i, the sentence i is "current" , the sentences before sentence i are "past" and the sentences after sentence i are "future". The *Timestep* layer then projects a paragraph to embeddings according to the timestep IDs of its sentences. This input representation method adopts a procedural manner to parse a paragraph, which is somehow similar to how humans read a paragraph. CGLI turns out to achieve much better performance in detecting the conflicting sentences and the physical states that lead to the conflicts than the baseline performance in the TRIP paper.

Jiang et al. (2023) brought up Learning from Experience by Annotating Procedures (LEAP), which follows the main structure of CGLI but utilizes the data augmentation technique more heavily compared with CGLI. LEAP uses multiple datasets as the source of data augmentation and processs these datasets with three methods including participant abstraction, word insertion and automatic labeling. The test results of LEAP also witnesses a huge gain in the performance of detecting conflict-

ing sentences and classifying the physical states, although data augmentation does not benefit the model a lot.

2.3 ART Dataset

Bhagavatula et al. (2020) produced the ART dataset to investigate the viability of language-based abductive reasoning. Specifically, abductive reasoning means inference from an incomplete set of observations to the most plausible explanation. The entire ART dataset is made up of around 20K commonsense narrative contexts with over 200K explanatory hypotheses. As shown in Figure 4, each example in the ART dataset contains a pair of observations and two hypothesis options. The pairs of observations are drawn from the ROCStories dataset (Mostafazadeh et al., 2016), which is a collection of short fivesentence stories. The beginning and ending for each story in the ROCStories dataset map to the first and second observations in each pair of observations in the ART dataset respectively. For the two hypothesis options, one is plausible while another is implausible. Both the plausible and implausible hypothesis options are crowdsourced on Amazon Mechanical Turk (AMT).

Bhagavatula et al. (2020) also finds that the ART dataset can not only serve as a benchmark, but it can also be used as a resource of transfer learning. By first traning a model on the ART dataset and then training on target commonsense datasets, the model will achieve better performance. In our project, we use the ART dataset for transfer learning purpose in Approach 1.

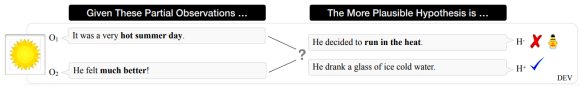

Figure 4: An example in the dev split of the ART dataset (Bhagavatula et al., 2020)

2.4 DistilBERT

DistilBERT is a streamlined version of BERT, a popular language model in natural language processing (NLP). The work by Sanh et al. from Hugging Face focuses on creating a model that retains the language understanding capabilities of BERT while being smaller, faster, and more efficient. This is particularly important in scenarios where computational resources are limited, such as on-edge devices or when operating under constrained computational budgets.(Sanh et al. (2019))

The key innovation of DistilBERT lies in its architecture and training process. DistilBERT mirrors the general architecture of BERT but is significantly smaller. This reduction in size is primarily achieved by halving the number of layers in the Transformer architecture, which is central to BERT. Interestingly, the research by Sanh et al. (2019) found that reducing the number of layers has a more substantial impact on computational efficiency than adjusting other dimensions, like the size of the hidden layers.

In DistilBERT, certain elements of BERT, such as the token-type embeddings and the pooler, are removed. The model is trained using knowledge distillation, a technique where a smaller model (student) is trained to replicate the behavior of a larger model (teacher). DistilBERT is initialized by taking layers from the teacher model (BERT) at intervals, effectively inheriting some of its knowledge. For training, DistilBERT uses the same corpus as BERT, which includes English Wikipedia and the Toronto Book Corpus. Despite its reduced size, DistilBERT manages to retain 97% of BERT's language understanding capabilities and operates 60% faster. This efficiency is a significant achievement, considering that the model is 40% smaller than BERT.

3 Approach

Our project mainly focuses on two approaches: 1) Transfer learning from the ART Dataset and 2) Fine-tune DistilBERT. In both approaches, we will use the same reasoning system structure (Figure 2) and the same four kinds of loss functions in the TRIP paper to fine-tune any pre-trained model. After getting the results of our approaches, we will conduct a comparative study between the performance of our approaches and the baseline performance in the TRIP paper.

3.1 Approach 1: Transfer Learning from ART

As ART is a much larger dataset than TRIP and it is also a commonsense reasoning dataset, transfer learning from ART may boost a model's performance on TRIP. Specifically, we will first fine-tune a pre-trained model on the ART dataset and then fine-tune this model on the TRIP dataset. For this approach, we will use **bert-base-uncased (denoted as BERT-bs below)** to be the pre-trained model for fine-tuning. Due to the limited computing resources we have, we choose bert-base-

Model	Accuracy (%)	Consistency (%)	Verifiability (%)
All losses			
BERT-bs (w/o ART)	73.79	1.14	0.28
BERT-bs (w/ ART)	**78.92**	0.85	0.28
DistilBERT	**78.92**	1.14	1.14
Omit Story Choice Loss			
BERT-bs (w/o ART)	72.93	**19.09**	3.7
BERT-bs (w/ ART)	71.79	18.80	4.27
DistilBERT	72.57	17.38	**8.63**
Omit Conflict Detection Loss			
BERT-bs (w/o ART)	41.60	0.00	0.00
BERT-bs (w/ ART)	42.45	0.00	0.00
DistilBERT	38.46	0.00	0.00
Omit State Classification Losses			
BERT-bs (w/o ART)	75.78	3.13	0.00
BERT-bs (w/ ART)	76.64	8.26	0.00
DistilBERT	70.37	3.13	0.00

Table 1: The evaluation result of our approaches.

uncased instead of bert-large-uncased which was used in the original TRIP paper. We will also fine-tune BERT-bs directly on TRIP to see whether the transfer learning from ART is helpful or not.

3.2 Appraoch 2: Fine-tune DistilBERT

Since the TRIP paper only evaluated the performance of BERT, RoBERTa, and DeBERTa, we try to use **DistilBERT** as an alternative model to see how it will perform in the same task. Although we do not expect that DistilBERT will result in better performance, it can be treated as a supplement to the TRIP paper. We choose DistilBERT mainly because it is relatively smaller, faster, and more efficient. For this approach, we will use **distilbert-based-uncased** to be the pre-trainied model for fine-tuning.

4 Evaluation Results

Table 1 shows the performance of all three models used in our approach (i.e. BERT-bs with transfer learning from ART, BERT-bs without transfer learning from ART, and DistilBERT). The performance is also measured in accuracy, consistency and verifiablity, which are used in the TRIP paper.

Hyperparameters	Accuracy (%)
bs=20, lr=1e-5	52.02
bs=20, lr=5e-5	53.13
bs=15, lr=1e-5	54.44
bs=15, lr=5e-5	**54.57**
bs=10, lr=1e-5	53.00
bs=10, lr=5e-5	52.74

Table 2: Result of fine-tuning BERT-bs on ART, where bs represents batch size and lr represents the learning rate of AdamW optimizer.

4.1 Transfer Learning from ART

Experiment on ART Since ART is a multiple choice dataset, we load BERT-bs from Huggingface API by using the AutoModelForMultipleChoice class. We randomly select 2000 examples from the training split and use the whole validation split for testing purpose. Before feed the training data into the model, we first parse and tokenize each example with the following format: [CLS] Observation1 Observation2 [SEP] [Hypothesis1 or Hypothesis2] [SEP]. We also write a customized data collator to pad sentences in each batch into the same length. To keep consistent with the TRIP paper, we use Adaw optimizer as the optimizer when training the model on ART. Then we fine-tune BERT-bs for 10 epochs and perform grid search on two different learning rates of AdamW and three different batch sizes. The results are given in Table 2. We save the model with the best accuracy (54.57%) and then fine-tune it on the TRIP dataset.

Experiment on TRIP We fine-tune and test both BERT-bs with transfer learning and Bert-bs without transfer learning by following exactly the same architecture in the TRIP paper (Figure 2). However, it turns out that BERT-bs with transfer learning does not significantly outperform BERT-bs without transfer learning. When all losses are considered, BERT-bs with transfer learning achieve an accuracy of 78.92%, which is $\sim$5% higher than BERT-bs without transfer learning. But when other three loss functions are used, the difference in the accuracy of these two models is only $\sim$1%. For consistency and verifiablity, the two models also achieve quite similar results. The consistency of BERT-bs with transfer learning is higher than the consistency of BERT-bs without transfer learning only when the

state classisfication loss is omitted. Both models perform poorly on verifiablity.

4.2 Fine-tune DistilBERT

We directly finetune DistilBERT on TRIP by still using the architecture in the TRIP paper (Figure 2). The testing results for DistilBERT are close to the results of both BERT-bs with transfer learning and BERT-bs without transfer learning. Although DistilBERT get the highest verifiability among all models in our experiment, its verifiability is still below 10%, which does not indicate any improvements on the baseline performance in the original TRIP paper.

4.3 Summary of Best Models

The best performance based on the three evaluation metrics and the corresponding models are listed below.

- **Best Accuracy**: 78.92% (All losses, both BERT-bs without transfer learning from ART and DistilBERT)
- **Best Consistency**: 19.09% (Omit story choice loss, BERT-bs with transfer learning from ART)
- **Best Verifiability**: 8.63% (Omit story choice loss, DistilBERT)

5 Discussion of Results

5.1 Effectiveness of Transfer Learning

We initially assume that the transfer learning can be useful because ART and TRIP are both commonsense reasoning datasets and ART can augment the number of examples for training given that TRIP is small. However, as mentioned in Section 4.1, the tiny difference in the performance of BERT-bs with transfer learning and BERT-bs without transfer learning is not sufficient enough to conclude that the transfer learning from the ART dataset is effective for boosting the performance on TRIP.

The ineffectiveness may be attributed to two reasons. First, the accuracy which BERT-bs achieves on ART is low, which is just several percent higher than 50%. ART itself is a challenging dataset and our accuracy is close to what Bhagavatula et al. (2020) got when the model is trained on 2000 examples. But such performance is barely better than a random guess, because there are only two hypothesis options in each example of ART. In other words, our model does not correctly learn from the ART dataset. Then it is likely that this model does not significantly differ from BERT-bs without transfer learning when it is fine-tuned on TRIP. Second, ART is still quite different from TRIP while we think they share some similarities. ART is designed for the abductive reasoning task. More specifically, this task focuses on how a language model can deduce correct explanation from a few observations/events, but it does not care about the sequence of these observations. In contrast, the examples in TRIP is composed of stories, In each story, there may be connections between the earlier sentences and the later sentences. Thus, the difference in ART and TRIP may also explain why transfer learning from ART is not helpful as expected.

5.2 Performance Pattern for Different Loss Functions

By observing the performance based on different loss functions, it can be found that the performance pattern of our approaches conforms to the pattern of baseline performance. A language model achieves higher accuracy when all losses are considered or state classification losses are omitted. Omitting story choice can make a model reach higher consistency and verifiablity. Omitting conflict detection loss results in zreo consistency and verifiablity. Omitting state classification losses maintains some consistency but no verifiablity. The detailed explanation for this pattern is as followings.

- **All Losses**: When the model is trained with all loss functions , it is being optimized for a comprehensive understanding of the story. This holistic approach generally leads to higher accuracy because the model is simultaneously fine-tuned on multiple aspects of the narrative.
- **Omit Story Choice Loss** : Removing the story choice loss from the training process shifts the model's focus away from directly determining the plausibility of entire stories. Instead, the model concentrates more on the internal consistency of the narratives. This shift can lead to an increase in consistency and verifiability as the model becomes better at understanding and predicting the physical states and conflicts within the story, even though it might be less accurate in the final story choice decision.

- **Omit State Classification Losses** : Excluding the losses for state classification likely makes the model less adept at understanding the specific physical states of entities within the narratives. However, it still retains a decent level of overall narrative understanding, leading to relatively high accuracy. This is because the model still attempts to detect conflicts and make story choices without the detailed comprehension of each entity's state.

- **Omit Conflict Detection Loss**: The loss for conflict detection is crucial for identifying inconsistencies within the narrative. When this is omitted, the model struggles to detect where the narrative breaks down in terms of physical plausibility. This deficiency significantly impacts the model's performance, leading to lower scores in accuracy, consistency, and verifiability, as it fails to grasp the critical elements that make a story implausible.

5.3 Indication to Original Reasoning System

Our approaches use BERT-bs and DistilBERT as the models for our experiments. These two models are much smaller than the baseline models used in the TRIP paper. In particular, DistilBERT only contains about 60% of BERT's data. Although it is generally presumed that the reduction in model size and data might lead to a notable decrease in performance, BERT-bs and DistilBERT achieve the results comparable to the outcomes of the TRIP paper. The best accuracy we achieve is even slightly greater than the best accuracy in the TRIP paper. The best verifiablity of our approaches and the best verifiablity of the TRIP paper are both around 10%. The difference in the best consistency of our approaches and the TRIP paper is less than 10%. This indicates that the non-ideal performance on TRIP in both our approaches and the TRIP paper may result from limitations in the original reasoning system structure (Figure 2). Fine-tuning pre-trained models through this structure is not quite helpful for models to understand and reason the contexts between sentences.

This insight aligns with findings from related work in which completely different structures were employed and significantly higher consistency and verifiability have been achieved. Ma et al. (2022) mentioned several sub-optimal design decisions in the TRIP paper's reasoning system structure. For example, the original structure does not consider pairs of sentences when detecting conflicting sentence pairs. Also, the structure uses the same encoded representations for both story choice classification and conflicting sentence detection. The fact that those newly proposed structures achieved high consistency and verifiability implies that the choice of model might not be as crucial as the design of the system's architecture in achieving optimal performance in tasks involving narrative understanding and reasoning.

6 Conclusion

In our project, we explore the possibilities of boosting the baseline performance on TRIP by implementing two approaches: 1) Transfer learning from ART and 2) Fine-tune an extra pre-trained model DistilBERT. Both strategies adhere to the reasoning system structure and loss functions outlined in the original TRIP paper.

The final results show that we do not make meaningful improvements on the baseline performance. The performance of the model with transfer learning does not significantly differ from the performance of the model without transfer learning. This may be caused by low accuracy the model achieves on ART and the difference between ART and TRIP. Besides, we discover that smaller models like BERT-bs and DistilBERT demonstrate comparable performance to the larger models in the TRIP paper. It is inferred that there could be potential issues with the original reasoning system structure. In other words, to enhance the performance on TRIP, the architecture used to fine-tune models might be more critical than the choice of the model itself. Future work should explore alternative architectures and training strategies, which we believe are the key factors to get better performance on TRIP.

7 Work Division

7.1 Team Composition

We form a team of two: Zhixuan Chen (CSE Master student) and Yichen Zhong (CSE Master student).

7.2 Project Timeline

1. Milestone I: Read the TRIP (Storks et al) and related paper thoroughly. This is due on Thu Nov 9

2. Milestone II: Reproduce the result for the Storks et al paper, including data manipu-

lation, model building, and evaluate performance of BERT, RoBERTa, and DeBERTa. This is due on Tue Nov 14

3. Milestone III: Apply other evaluation algorithms such as DistilBert to the TRIP dataset and compare results. This is due on Fri Nov 17
4. Milestone IV: Apply transfer learning methods, i.e., first fine tune the model by using the ART dataset and then feed the TRIP benchmark. This is due on Wed Nov 22
5. Milestone V: Compare and evaluate the result using different metrics, summarize the findings and prepare for final presentation. This is due on Fri Nov 24
6. Milestone VI: Write the final paper report. This is due on Wed Dec 6

7.3 Zhixuan Chen's Contribution

1. Reproduced TRIP results
2. Fixed pre-processing errors when loading the TRIP dataset
3. Fine-tuned the ART dataset and applied Transfer Learning to TRIP
4. Trained the BERT-base-uncased with TRIP (with ART finetuned)
5. Compared results with the baseline and draw some conclusions
6. Wrote Introduction, Related work (Recent Progress, the ART Dataset), Approach (Transfer Learning part), Evaluation Results (Result for Transfer Learning with ART), and Discussion parts of the report

7.4 Yichen Zhong's Contribution

1. Reproduced TRIP results
2. Trained BERT-base-uncased with TRIP (without ART finetuned)
3. Trained DistilBert with TRIP (without ART finetuned)
4. Compared results with the baseline and draw some conclusions
5. Wrote Abstract, Related work (The TRIP Dataset, DistilBert), Approach (Evaluate with DistilBert part), Evaluation Results (Result for Evaluating with DistilBert), Discusssion, and Conclusion part.
6. Setup GitHub Repo and Made Presentation Slides

8 URL Link to Codebase

Here is the link to our code on GitHub.

References

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Scott Wen tau Yih, and Yejin Choi. 2020. Abductive commonsense reasoning.

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.

Aditya Gupta and Greg Durrett. 2019. Effective use of transformer networks for entity tracking. 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 2019)*, Hong Kong, China. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention.

Yifan Jiang, Filip Ilievski, and Kaixin Ma. 2023. Transferring procedural knowledge across commonsense tasks.

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.

Kaixin Ma, Filip Ilievski, Jonathan Francis, Eric Nyberg, and Alessandro Oltramari. 2022. Coalescing global and local information for procedural text understanding. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1534–1545, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4658–4664, Florence, Italy. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. Accepted at the 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing - NeurIPS 2019.

Shane Storks, Qiaozi Gao, Yichi Zhang, and Joyce Chai. 2021. Tiered reasoning for intuitive physics: Toward verifiable commonsense language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4902–4918, Punta Cana, Dominican Republic. Association for Computational Linguistics.