EECS595 Final Project
Improving the Baseline Performance of the TRIP Model

Wenfei Tang
twenfei@umich.edu

Juejue Wang
juejuew@umich.edu

Abstract

Observing the unsatisfactory baseline performance of large-scaled language models on Tiered Reasoning for Intuitive Physics (`TRIP`), a newly proposed commonsense reasoning dataset, we propose to perform architecture modifications and optimization schedules of transfer learning as potential methods for improvement.

We have done experiments on training the `TRIP` model with transfer learning by pre-training on eight different datasets, ranging from question-answering to inference tasks. All of these intermediate tasks emphasize on the model's reasoning ability. We also explored efficient-parameter tuning by adding an adapter module to the `RoBERTa-base` transformer and compare its performance with other fine-tuning methods.

1 Introduction

In recent years, researchers have developed dozens of large-scale benchmark datasets to capture physical or scientific reasoning for Natural Language Processing (NLP). Existing benchmarks typically suffer bias, especially when dealing with high-level benchmark tasks where systems may pass over reasoning and give unjustified prediction with artificially high accuracy (McCoy et al., 2019; Belinkov et al., 2019).

In recognition of this issue and to better measure the machine's ability in understanding and reasoning physical commonsense, Storks et al. (2021) introduced an unprecedented dataset `TRIP` together with three metrics. Their coherent reasoning chain was built from low-level to high-level tasks thus not only enabling the evaluation in a human-interpretable sense but also alleviating issues concerning data bias to some extent.

However, a tiered baseline for `TRIP` demonstrates a low performance of existing language models with verifiability of 10.8% on the proposed joint tasks. This shows that large-scale language models with huge size of pre-training data (e.g., BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019), DeBERTa (He et al., 2020)) struggle to perform tiered reasoning tasks despite having high accuracy if applied to the end task directly. The goal of our project is thus to develop approaches to improve the baseline performance of the various large-scaled language models on `TRIP`.

Classic supervised learning accomplishes training in an isolated environment on a single dataset. Transfer learning, however, allows us to train a model on a series of datasets of additional domains or tasks, and it has been proven beneficial in improving the performance of many predicative language models in the Natural Language Processing field (Ruder et al., 2019).

In this project, we performed architecture modifications and optimization schedules (Ruder et al., 2019) to improve the performance baseline of `BERT`, `RoBERTa` and `DeBERTa` on `TRIP`. Before fine-tuning the pretrained model on the target task, we added another layer of training the models on a relevant intermediate task to improve the baseline performance of these models. We adapted the multi-tiered quantitative evaluation of commonsense reasoning proposed for `TRIP`, which uses accuracy, consistency, and verifiability as evaluation metrics. We focused on consistency and verifiability to measure the low-level predictions in the reasoning process.

The rest of this report is organized as follows: Section 2 explains details about transfer learning and adapters and introduces several relevant benchmark datasets. Section 3 describes the target and intermediate tasks and the transfer learning we experimented with. Section 4 summaries the performance of our proposed approaches. Section 5 discusses the limitations and contributions of our project and elaborates on future work. The last

Name	Task	Domain/ Source	Metrics
Sequence classification			
BoolQ (Clark et al., 2019)	binary QA	Wikipedia, web queries	acc.
Multiple-choice			
Hellaswag (Zellers et al., 2019)	commonsense-reasoning	misc.	acc.
CosmosQA (Huang et al., 2019)	commonsense reasoning	crowdsourced	acc.
PIQA (Bisk et al., 2020)	commonsense reasoning	misc.	acc.
ARC (Aristo) (Clark et al., 2018)	multiple-choice QA	misc.	acc.
RACE (Lai et al., 2017)	reading comprehension	English exams	acc.
WinoGrande (Sakaguchi et al., 2020)	coreference resolution	crowdsourced	acc.
ART (Bhagavatula et al., 2019)	NLI	stories	acc.

Table 1: Overview of intermediate tasks used in our experiments, grouped by task type.

two Sections conclude our findings and provide information about the division of work.

2 Related Work

Transfer Learning

Transfer learning is a technique that uses deep learning models trained on a large dataset to perform similar tasks on another dataset. Sequential transfer learning has two phases: a pretraining phase on a source task, and an adaptation phase that applies the learned knowledge to a target task. The adaptation phase of transfer learning has two major methods: architecture modifications and optimization schedules (Ruder et al., 2019). Architecture modifications include changing the number of embeddings, layers, modules, and other architecture inside the pretrained model. Optimization schedules include fine-tuning part of the pre-trained model and fine-tuning the pre-trained model on a series of datasets and tasks.

Fine-tuning is one of the most common transfer learning techniques used in NLP (Houlsby et al., 2019). It copies the weights from a pre-trained network on an intermediate task and tunes this network on the downstream or target task. Recent work has shown that fine-tuning usually enjoys a good performance and leads to a transfer gain.

However, transfer learning does not guarantee a transfer gain. According to the results of (Poth et al., 2021) which experimented on a wide range of task combinations for RoBERTa, 243 (53%) transfer combinations yield positive transfer gains whereas 203 (44%) yield losses. Therefore, the significance of identifying the right datasets to pre-train on is highlighted.

Adapter Modules

Adapter tuning is a parameter-efficient way to perform transfer learning without fining tuning the entire model proposed by Houlsby et al. (2019). A bottleneck adapter module consisting of a small number of new parameters is added to the model, and only the new adapter top-layer will be trained while the original network's parameters remain unchanged. In this way, parameters are shared between the original and new network to a great extent and there is no need to train an entirely new model. It is first used under the online setting where the same network is reused for the training of multiple downstream tasks.

See 1 for the adapter's architecture (Houlsby et al., 2019). Two adapter modules are added to the Transformer Layer. Each adapter Layer consists of the layers shown on the right architecture. In the fine-tuning phase of transfer learning, only the green layers are trained on the downstream task, while the parameters from the original network remain the same.

TRIP dataset

Recent work has shown that large-scale language models lack verifiable reasoning despite having high accuracy on the end task. Large-scale benchmark datasets targeting commonsense reasoning tasks (e.g., (Mishra et al., 2018), (Bisk et al., 2020)) typically do not support the evaluation of the reasoning process. To address this problem, a new benchmark dataset `TRIP` is introduced (Storks et al., 2021). It uses story plausibility classification as the end task and has dense annotations for capturing multi-tiered reasoning. Models with satisfactory performance on previous datasets may fail the tasks posed by `TRIP`, because `TRIP` emphasizes

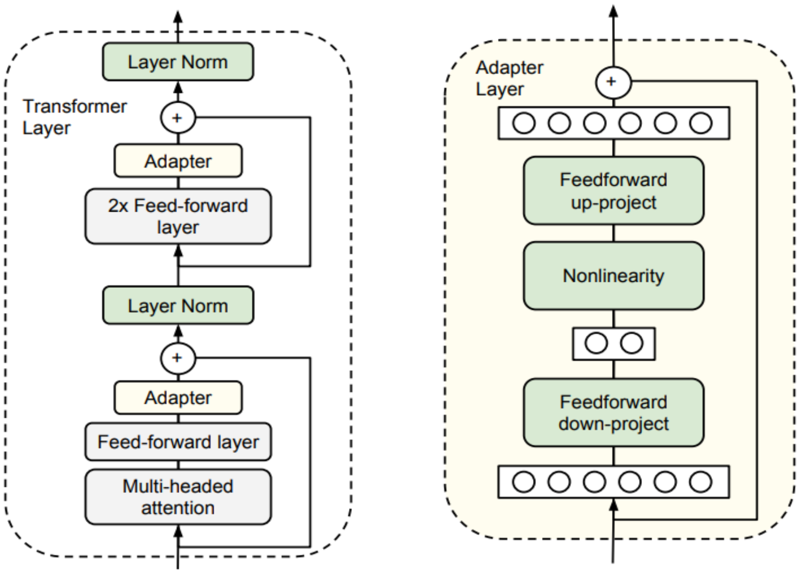

Figure 1: Architecture of the adapter module and how it fits in the Transformer model (Houlsby et al., 2019). Only the green layers are trained on the downstream task.

the language model's verifiable physical common sense reasoning ability. A set of physical states and new metrics, verifiability, and consistency are also introduced to measure the language model's tiered reasoning ability. Language models are evaluated on verifiability and consistency, where verifiability evaluates the model's ability to detect the change of physical states and how it affects the plausibility of the story and consistency evaluates the model's ability to detect conflicts in the story.

Many of the large-scale language models, though having high accuracy on the end task, fail to achieve high consistency and verifiability and consistency on the `TRIP` dataset. With experiments done on state-of-art popular language models, the highest consistency is only 28.0%, achieved by BERT, and the highest verifiability is only 10.6%, achieved by RoBERTa.

PIQA dataset

Introduced in 2020, Physical Interaction: Question Answering (`PIQA`) is a new benchmark dataset for physical commonsense reasoning (Bisk et al., 2020). The modeling of physical commonsense knowledge places a challenge on AI's ability in interacting with the physical world. This is essential especially for the development of robots that understand and respond to natural languages. Recent progress has been made on abstract tasks through large-scale pretraining models, while whether these models can capture physical commonsense knowledge remains unclear. `PIQA` is thus introduced to fix this gap.

Covering the wide aspects of phenomena, the `PIQA` benchmark requires the capture of the knowledge of basic properties of the objects, as well as the correct identification of more preferable answers, which requires high-level commonsense reasoning. The accuracy achieved by human is about 95%, while the large-scale pretrained model struggles with this task and achieves the highest accuracy of about 77%. The physical common sensing reasoning of `PIQA` shares similarity with the `TRIP` task.

Hellaswag dataset

The `Hellaswag` dataset (Zellers et al., 2019) is introduced to answer the question: "Can machine perform human-level commonsense inference despite reaching human-level performance with respect to evaluation metrics?" The sources of this dataset include video captions from the ActivityNet Captions dataset (Krishna et al., 2017) and an online how-to manual, WikiHow.

By requiring machines to choose the most reasonable followup for an event description, this task measures the commonsense reasoning ability of state-of-the-art models. Evaluation results suggest that humans find the task easy and achieve an accuracy that is greater than 95%, but state-of-the-art models struggle with this task with an accuracy of less than 48%.

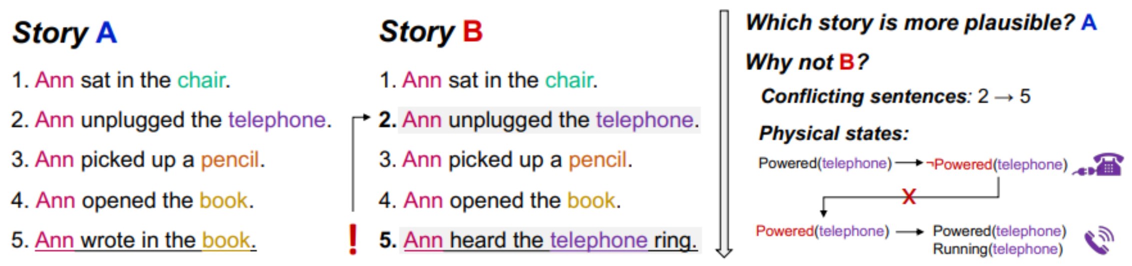

Figure 2: A example of story pairs from the TRIP dataset (Storks et al., 2021), with conflicting pairs and change of physical states.

3 Approaches

Target Task

For the target task, we are going to use the TRIP dataset and follow the proposed tiered reasoning system. Our goal is to improve the baseline performance of the TRIP model.

Intermediate Task

Recent work has shown that inference tasks and commonsense reasoning QA tasks are generally useful as intermediate tasks (Pruksachatkun et al., 2020). MNLI and CosmosQA are proven to be generally helpful in increasing the performance of the target task. Moreover, since we use TRIP dataset as the downstream task, intermediate tasks should be chosen based on their similarity with the TRIP dataset. The intermediate tasks should emphasize the model's reasoning abilities, and preferably be a question-answering or an inference task.

Based on this rule, the intermediate tasks we explored include CosmosQA (Huang et al., 2019), BoolQ (Clark et al., 2019), Aristo (Clark et al., 2018), Hellaswag (Zellers et al., 2019), RACE (Lai et al., 2017), WinoGrande (Sakaguchi et al., 2020), ART (Bhagavatula et al., 2019), and PIQA (Bisk et al., 2020). Detailed information for each task is summarized in table 1.

Fine-tuning

Three different transformers are considered in our project: RoBERTa-base, RoBERTa-large, and GPT.

In the fine-tuning phase, we copy the weights from the pre-trained transformers on eight different datasets and use them as the starting point for the training of the TRIP model.

Adapter Tuning

We performed architectural modifications by adding an adapter module on top of the transformers. Adapter tuning is first proposed to be used on a list of downstream tasks (Houlsby et al., 2019), and we applied it in our project to make transfer learning more parameter-efficient given our limited resourced on Great Lakes. Instead of fine-tuning 110M of parameters on RoBERTa-base, we used adapters pre-trained on ART and CosmosQA as the pre-trained networks and added another layer of adapters with a small number of parameters. During the training process, we leave the parameters of RoBERTa-based untouched, and only train on the adapter layer. There is no doubt that the performance of such models will be worse than models trained with RoBERTa-large with 1.5B parameters, but our results show that after applying adapters, the model with transfer learning obtained almost similar performance with models trained on RoBERTa-large with faster training, indicating a satisfactory trade-off between performance and time.

Optimizer Selection

With the baseline TRIP with no transfer learning, we also perform experiments and compare the performance of the following optimizers: AdamW (proposed in the original TRIP paper), Adam, and SGD.

4 Evaluation

Evaluation Metrics

Accuracy, Consistency and Verifiability

Accuracy is used to measure the end task prediction performance and is calculated by dividing the total number of testing examples by the number of

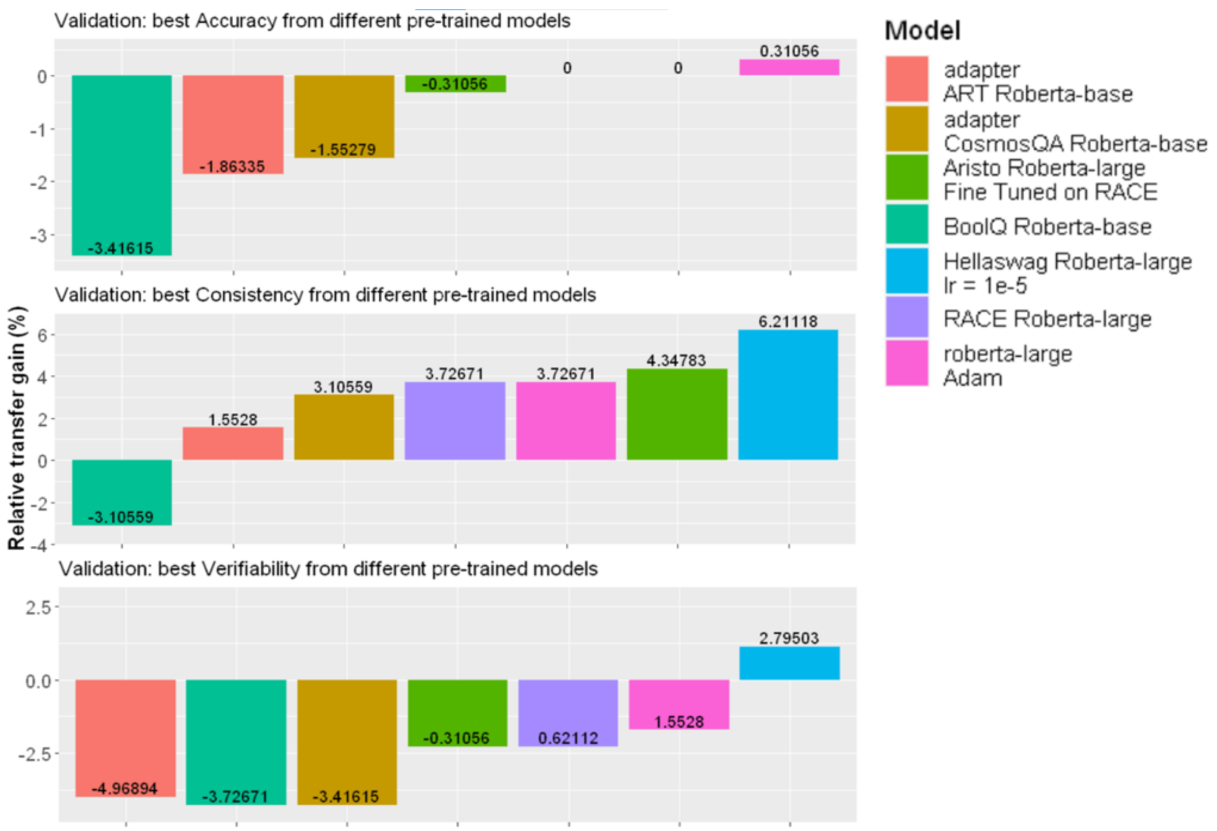

Figure 3: Transfer gains are calculated against the best baseline `TRIP` model with no transfer. Models are evaluated on the **validation/dev dataset** of `TRIP` with accuracy, consistency, and verifiability.

stories that are correctly classified. Based on the accuracy, **consistency** further measures the ability to identify conflicting sentence pairs for implausible stories. On the basis of consistency, **verifiability** further assesses the ability to correctly identify the underlying physical states which cause the conflict.

Model Performance

We visualized the metrics values in Figure 5. The baseline accuracy, consistency, and verifiability on the validation dataset are approximately 76.7%, 22.0%, and 9.6% respectively; the baseline accuracy, consistency, and verifiability on the test dataset are 77.5%, 25.4%, and 8.5% respectively.

On the validation set, `Hellaswag` produces the best performance on consistency and verifiability compared with other models. The consistency is about 6% better than the baseline, and the verifiability is about 3% better than the baseline. `RACE` obtains the highest accuracy of 77.0%, which is close to the performance of the baseline.

On the test set, `Hellaswag` obtains the highest consistency of 25.9% and the highest verifiability of 9.7%. For accuracy, `ART` performs the best with an accuracy of 78.6%. However, `ART` performs significantly worse than the baseline in terms of consistency and verifiability.

Transfer Gains and Losses

We created multiple plots to visualize the transfer gains and losses. Figure 3 summarizes the performances of different pretrained models on the validation dataset. Although the improvement in accuracy and verifiability is tiny, with high consistency, most models outperform the baseline in detecting conflicts in the story. In general, `Hellaswag` is the model the performs best on the validation dataset.

We also compare the performances of the models on the test dataset, using Figure 4. The results are generally consistent with the results on the validation dataset. According to Figure 4, `RoBERTa-large` pretrained on `Hellaswag` shows the highest consistency and verifiability. `RoBERTa-large` pretrained on `Aristo` and `RACE` also produce reasonably high values of metrics. These models performs better than the baseline when considering consistency and verifiability. This may imply potential advantages of transfer learning in tiered commonsense reasoning.

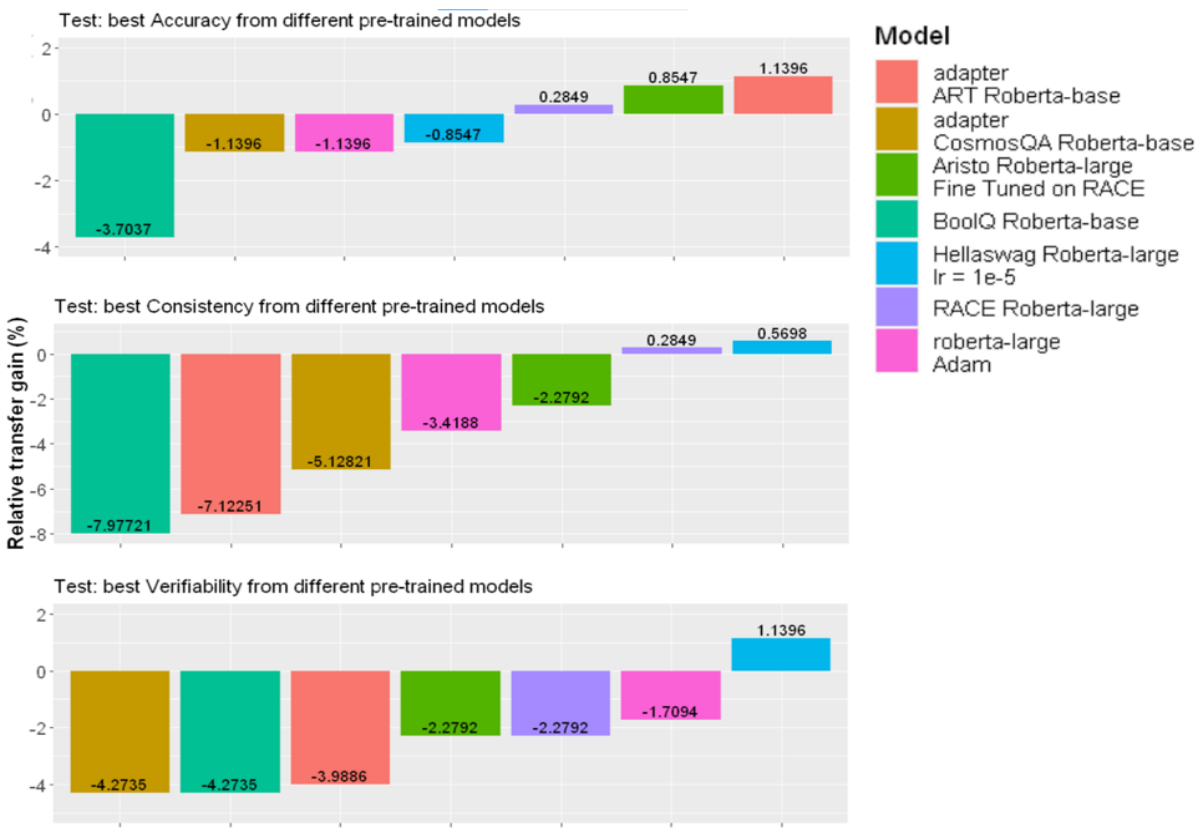

Figure 4: Transfer gains are calculated against the best baseline `TRIP` model with no transfer. Models are evaluated on the **test dataset** of `TRIP` with accuracy, consistency, and verifiability.

However, models based on `RoBERTa-base` generally performs worse than the the baseline. `RoBERTa-large` is a larger model based off `RoBERTa-base`, which may explain the relatively worse performance of the `RoBERTa-base` models. Moreover, transfer learning still shows its potential in improving the baseline performance as the differences in values of metrics is reasonably small.

5 Discussion

Models with more Parameters

We provided a plot for average relative gains obtained by applying transfer learning (see Figure 6). In this figure, we divided the models into two categories: using `RoBERTa-base` with adapters, and using `RoBERTa-large` with fully fine-tuning parameters . We can see that the transfer gains are all much higher if we use `RoBERTa-large` with fully fine-tuning parameters. Below are the two reasons of such results. Firstly, `RoBERTa-large` with fully fine-tuning parameters will train the entire network, while `RoBERTa-base` with adapters will only train the adapter module. Secondly, `RoBERTa-base` with adapters uses `RoBERTa-base` with 110M of parameters and `RoBERTa-large` has 1.5B of parameters.

Fine-tuning in Sequence

We also did some experiements on fine-tuning in sequence. There is one model we perform experiments on with a `RoBERTa-large` model pretrained on `RACE` and then transfer it to `Aristo`. After these two layers of transfer learning, we applied it again to `TRIP`.

Challenges

Reproducing the `TRIP` model is a great challenge to us because of our limited accessibility to high-performance GPUs. Adapters are introduced to our project when we need to perform a more efficient way for parameter-tuning given the limited time and resources. Setting up and resolving the Great Lakes also takes another huge chunk of time for us, because many of the issues are related to the underlying architecture of Great Lakes and some necessary packages cannot be installed properly.

Another challenge we would like to address is the selection of intermediate tasks. Given the fact

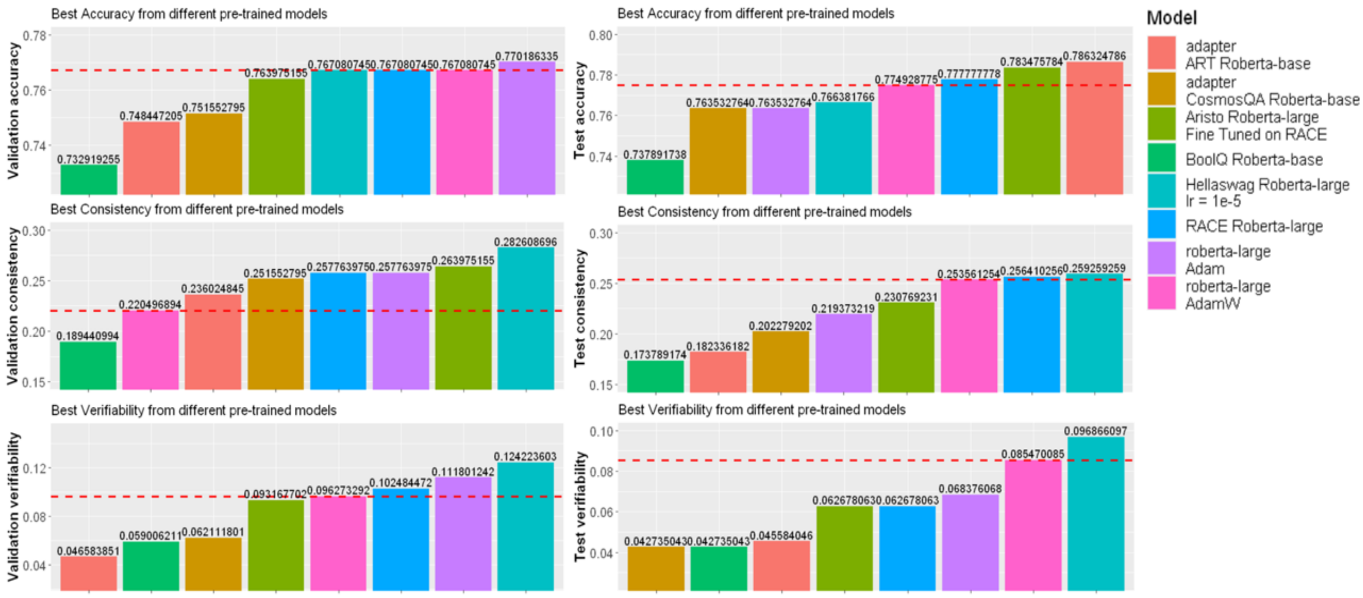

Figure 5: Performance of all models evaluated on TRIP validation and test datasets. Evaluation metrics are accuracy, consistency, and verifiablity. The red dotted line marks the best baseline performance of TRIP with no transfer learning.

that there is no guarantee of transfer gains on any one of the intermediate tasks, we have decided to move forward with a wide range of datasets and see how they affect the final performance of the TRIP model. This decision, though time-consuming, is proved to be the right choice given the experiment results we obtained–we did see some transfer losses, especially on the test datasets of TRIP, when performing transfer learning.

Limitations and Future Work

However, there are certain limitations to our work. The most outstanding limitation of our work is that we did not perform a comprehensive set of parameter combinations when performing grid-search. Given the limited resources we have, we could only run one set of parameters on each run, and performing an exhaustive set of combinations will be too time-consuming for us. Therefore, we only do one or two sets of parameters for each model after transfer learning and re-use the best combination of parameters from the TRIP model with no transfer learning.

We have refined the code base for training with a GPT-neo transformer. However, we were not able to run the training because of the limitation on CUDA memory.

We also attempted to train PIQA with RoBERTa-large, and because of the same reasons above, we could not finish our training.

If we have access to a better GPU, we would like to train TRIP with transfer learning and perform an exhaustive search on the parameter combinations. This should yield a more accurate result of which task serves the best as the intermediate task.

6 Conclusion

In this project, we explored transfer learning to improve the baseline performance of TRIP. We successfully implemented eight datasets, all of which have an emphasis on the language model's reasoning ability thus having great potential on the TRIP task. In addition, we experimented with parameter tuning through the adoption of an adapter module. We have also altered the learning rate and changed the optimizer to measure the effects.

Our results show that Hellaswag performs better than the baseline with respect to consistency and verifiability on the test and validation dataset. Aristo under two layers of transfer learning (a RoBERTa-large model pre-trained on RACE and then transfer it to Aristo) and RACE have comparable performance with the baseline. These findings may indicate some potentials of transfer learning in improving the tiered commonsense reasoning of large language models. Other models based on RoBERTa-base perform worse than the baseline and the relative simplicity of the model may explain this deficiency.

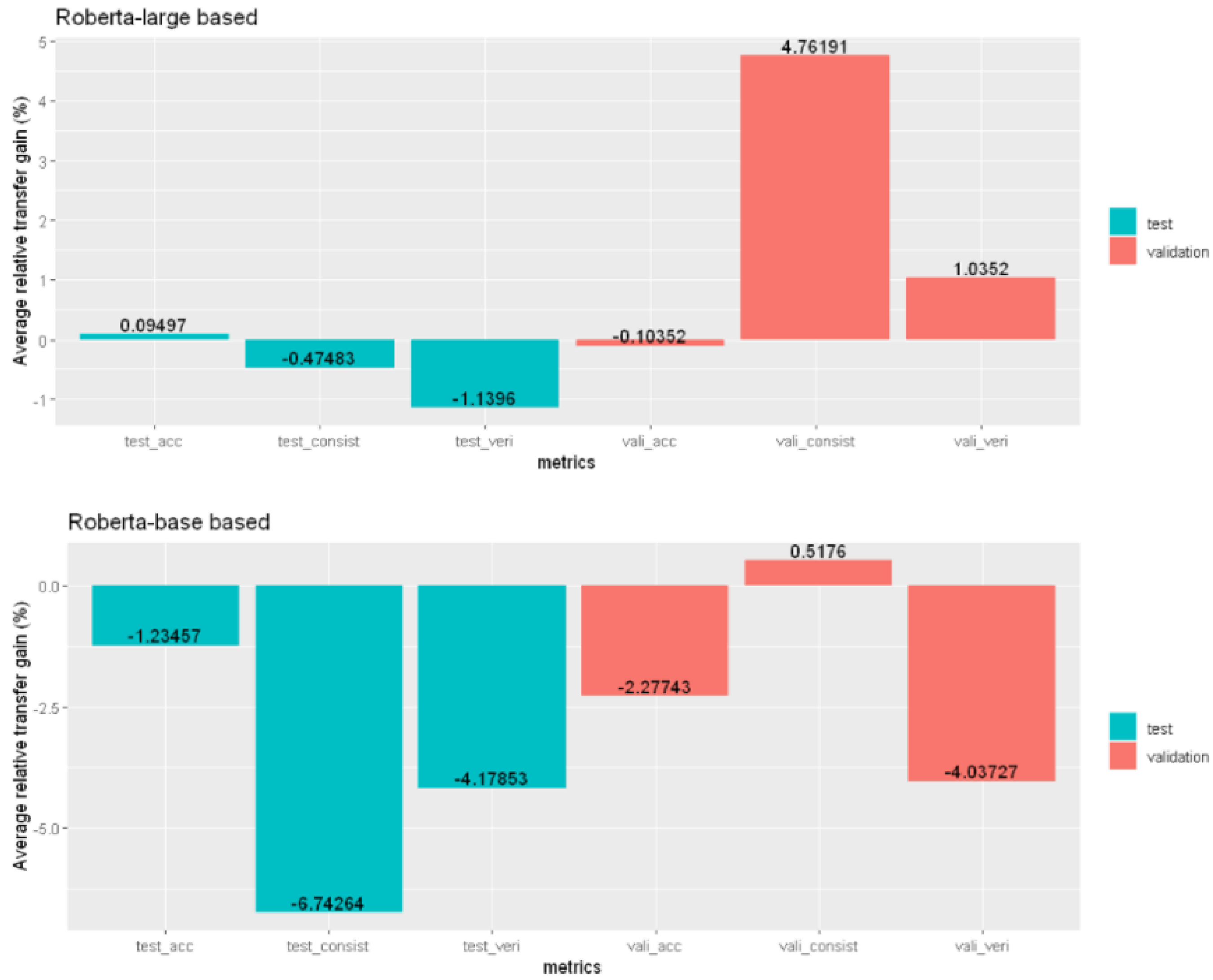

Figure 6: Average relative transfer gains obtained by applying transfer learning. The above figure shows the transfer gain of using roberta-large pre-trained models, and the below figure shows the transfer gain of using roberta-base pre-trained models. Red denotes the performance on the test dataset of TRIP, and blue denotes the performance on the validation dataset of TRIP. Model is evaluated on metrics of accuracy, consistency and verifiability.

7 Division of Work

Team Composition

We form a team of two: Wenfei Tang (major in CSE) and Juejue Wang (major in applied statistics).

Project Timeline

- 11/9/2021 - 11/15/2021: Collected feedback from Prof. Chai and GSIs to make sure the approaches and datasets are appropriate; modified the proposal based on the suggestions;
- 11/20/2021 - 11/26/2021: Got ourselves familiar with the code base of TRIP. Failed to use Google Colab to reproduce the results because of CUDA memory limitations on Colab.
- 11/27/2021 - 11/30/2021: Got ourselves familiar with running the code base on Great Lakes server. Successfully reproduced the results;
- 12/1/2021 - 12/9/2021: Start running experiments;
- 12/9/2021: Made slides and prepared for presentation;
- 12/10/2021 Project presentation; Collected questions and answered them;
- 12/11/2021 - 12/16/2021: Compose the project final report;
- 12/16/2021: Submit the final report and code;

Wenfei Tang's Contribution

1. Reproduced results from `TRIP`;
2. Fixed reproducing errors and some issues in the `TRIP` pre-processing code;
3. Modified the `TRIP` code base to allow transfer learning; added to the code to make it compatible with training `RoBERTa-base`, `GPT-NEO`, and `PIQA`;
4. Trained models on `Aristo`, `RACE`, `Wino-Grande`, `ART`;
5. Wrote the related work, approaches, and discussion parts of the report;

Juejue Wang's Contribution

1. Attemped to reproduce `TRIP` on Google Colab, and fixed environment issues of running `TRIP` on Great Lakes;
2. Reproduced results from `TRIP`;
3. Trained models on `BoolQ`, `Aristo`, `Hel-laswag`, `RACE`;
4. Wrote the introduction, evaluation and conclusion of the report;
5. Performed data analysis and made tables for the report;

8 Code Repo

Github Repo.

References

Yonatan Belinkov, Adam Poliak, Stuart M Shieber, Benjamin Van Durme, and Alexander M Rush. 2019. Don't take the premise for granted: Mitigating artifacts in natural language inference. *arXiv preprint arXiv:1907.04380*.

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Scott Wen-tau Yih, and Yejin Choi. 2019. Abductive commonsense reasoning. *arXiv preprint arXiv:1908.05739*.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7432–7439.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos qa: Machine reading comprehension with contextual commonsense reasoning. *arXiv preprint arXiv:1909.00277*.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, pages 706–715.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

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. *arXiv preprint arXiv:1907.11692*.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*.

Bhavana Dalvi Mishra, 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. *arXiv preprint arXiv:1805.06975*.

Clifton Poth, Jonas Pfeiffer, Andreas Rücklé, and Iryna Gurevych. 2021. What to pre-train on? efficient intermediate task selection. *arXiv preprint arXiv:2104.08247*.

Yada Pruksachatkun, Jason Phang, Haokun Liu, Phu Mon Htut, Xiaoyi Zhang, Richard Yuanzhe Pang, Clara Vania, Katharina Kann, and Samuel R. Bowman. 2020. Intermediate-task transfer learning with pretrained language models: When and why does it work? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5231–5247, Online. Association for Computational Linguistics.

Sebastian Ruder, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer learning in natural language processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18, Minneapolis, Minnesota. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8732–8740.

Shane Storks, Qiaozi Gao, Yichi Zhang, and Joyce Chai. 2021. Tiered reasoning for intuitive physics: Toward verifiable commonsense language understanding. *arXiv preprint arXiv:2109.04947*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.