Question Mastery Unleashed: Self-Generated Data Augmentation is All You Need for AI to Ace Exams

Aydan Huang, Viola Xu, Tian Zhou
Johns Hopkins University

1 Motivation

The more data we have, the better performance we can achieve. However, it is very too luxurious to annotate a large amount of training data. Therefore, proper data augmentation is useful to boost up model performance. Data augmentation has shown promising results in computer vision. Image can be augmented easily by flipping, adding salt, etc, and it has been proved that augmentation is one of the anchors to the success of computer vision models.

However, in the field of natural language processing, it is hard to augment text due to the high complexity of language: not every word we can replace it with others such as a, an, the. Also, not every word has a synonym. Even changing a word, the context will be totally different.

Motivated by the Self-instruct[1] framework, we aim to refine language models using self-generated instances. These techniques are often constrained by the inherent complexities of language, where straightforward substitutions or the use of synonyms may not be contextually viable. By utilizing self-generated content, our approach seeks to reduce the need for manually labeling and annotating large datasets, enhancing both task-specific and general capabilities of LMs. This strategy heralds a shift towards more efficient and effective training methodologies in NLP, potentially setting a new standard for model refinement.

2 Related Work

Self-Instruct: Aligning Language Models with Self-Generated Instructions [1] by Wang et al, introduces a framework to improve instruction-tuning's capabilities in large language models by generating new instruction data from the task seeds. The process prompts the model to generate new tasks from seeds with the corresponding input-output examples, and filters out low-quality or similar instructions. The model performs instruction-tuning on this generated dataset. This method resulted in significant improvements in following instructions when applied to GPT-3 model, demonstrating the utility of self-generated instruction data for enhancing language model performance without relying on extensive labeled datasets.

3 Experiment

3.1 Pipeline

To develop the proposed pipeline, we start by selecting 200 questions from the SAT or a similar test as seeds. We then apply the bootstrapping algorithm. The seeds are input into a language model (LM) to generate new question stems. Following this, we filter the generated questions to ensure quality and relevance. The LM is then tasked with creating answers for these filtered questions, which undergo a second round of filtering. This refined set of question-and-answer instances is used to further train the LM, with the goal of enhancing its ability to generate and solve exam-style questions. Through iterative refinement and training, we aim to build a dataset that significantly improves the

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

LM's performance in exam problem-solving. We then apply this dataset to fine-tune the model and examine its ability on the validation set.

Detail In the pipeline for instructing generations of question stems, we will design a template of instructions to guide this process. Based on the previous study[2], diversifying the data can enhance the generalization ability of LLMs, our template includes a variety of instructions such as "rewrite a given question into a more complex version with additional reasoning steps." This method aims to enrich the dataset, making it more robust and effective in training LLMs.

As for the filtering process, we plan to utilize REV[3] to sift through the generated questions and answers. Through REV evaluation, we filtered out a fixed portion of questions and answers with low score after each iteration, thereby streamlining the refinement process and enhancing the overall quality of our dataset.

When instructing LMs to generate answers, we will format our instructions to guide the LMs and lead to better accuracy. We consider adopting the methods of existing study such as Plan-and-Solve method[4].

3.2 Benchmarks and Distillation Baseline

We will use the data generated from our pipeline to fine tuning LMs, such as vanilla GPT3 or Llama 2-Chat 7B. We will compare our fine-tuned model with other LMs such as GPT3, InstructGPT[5].

Considering we may lack computation resources, we also consider setting benchmarks on smaller LMs such as GPT2 or DistillBert.

We plan to evaluate the model's performance on answering SAT or similar test datasets, and then we evaluate the accuracy. We can also test on other exam related dataset such as OpenbookQA.

3.3 Ablation Study

(a) Seed selection. The selection of seeds may bias the generated instances, further impacting the effect of fine-tuning. To explore the biases, we will generate instances and train the model from seeds that vary in difficulty, question category, linguistic variation, etc., and observe their impact on the model.

(b) Incorporating other domains. We aim to explore how the performance of the model on questions in a particular domain is affected by its knowledge in other domains. In addition to reasoning and math questions, we could also incorporate questions from other domains, for instance, history and programming questions, and evaluate the performance on models.

4 Dataset

The dataset consists of 10 SAT practice exams for question seeds and an additional 10 for evaluation purposes. From the initial 10 exams, we will extract 200 reasoning and math questions as seed prompt, and expand the seeds into 20000 instances of questions and answers for fine-tuning through the pipeline described above. Following this, the other 10 practice exams are used to evaluate the accuracy on the tuned model.

5 Hypothesis & Halfway Milestone

By incorporating the concept of fine-tuning with data augmentation from domain-specific seeds, we expect an improvement in the model's accuracy on SAT reasoning and mathematical tasks, and outperforming the general-purpose pretrained model (vanilla GPT-2, DistillBert, etc).

In our halfway milestone, we will implement bootstrapping algorithm capable of expanding the seeds, establish a filtering mechanism, and conduct a preliminary testing on instance generation, and examine the accuracy of the model trained on a particular question type (namely SAT reading reasoning question). After completing the milestone, we can refine the question generation process, filtering process, and generalize the model to questions in different domains.

References

References follow the acknowledgments in the camera-ready paper. Use unnumbered first-level heading for the references. Any choice of citation style is acceptable as long as you are consistent. It is permissible to reduce the font size to `small` (9 point) when listing the references. Note that the Reference section does not count towards the page limit.

[1] Wang et al., "Self-Instruct: Aligning Language Models with Self-Generated Instructions." https://arxiv.org/pdf/2212.10560.pdf

[2] YuLan-Chat-Team. 2023. Yulan-chat: An open- source bilingual chatbot. https://github. com/RUC-GSAI/YuLan-Chat.

[3] Chen, H., Brahman, F., Ren, X., Ji, Y., Choi, Y. and Swayamdipta, S., 2022. REV: information-theoretic evaluation of free-text rationales. arXiv preprint arXiv:2210.04982.

[4] Wang, L., Xu, W., Lan, Y., Hu, Z., Lan, Y., Lee, R. K.-W., & Lim, E.-P. (2023). Plan-and-Solve Prompting: Improving Zero-Shot Chain-of-Thought Reasoning by Large Language Models. arXiv preprint arXiv:2305.04091. Retrieved from https://arxiv.org/abs/2305.04091

[5] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Car- roll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training Language Models to Follow Instruc- tions with Human Feedback. In Advances in Neural Information Processing Systems (NeurIPS).