Modern Natural Language Processing (MNLP) - Progress Report

Camille Nicole Gisèle Challier | 311020 | camille.challier@epfl.ch
Céline Kalbermatten | 328519 | celine.kalbermatten@epfl.ch
Wesley Elliott Stephen Monteith-Finas | 324745 | wesley.monteith-finas@epfl.ch
Chatbots-R-Us

1 Introduction

This progress report provides in-depth analyses of the datasets, the performance evaluation metrics, the model choice and the plans to quantize the model. The two-staged training methodology used, including Supervised Fine-Tuning (SFT) followed by Direct Preference Optimization (DPO) to align to human-preferred answers, is also discussed. From these findings, a **fine-tuned Flan-T5 large model** appears to outperform the rest.

2 Datasets

2.1 Supervised Fine Tuning (SFT)

To begin with, SFT on domain-specific questions was applied using two datasets. The **Custom EPFL courses** dataset was created by extracting questions from exercises and exams from 5 courses (A.1.1) and includes 551 open questions and detailed MCQs answers. The **StemQ** dataset (Drori et al., 2023) consists of 441 open questions from 18 of the 27 curated STEM university courses.

2.2 Direct Preference Optimization (DPO)

Three datasets related to human preference learning from relevant domains were used for DPO training (Table 1).

Dataset Name	Train samples	Test samples
Milestone 1 (M1)	21'101	2'345
WebGPT Comparisons	12'911	1'435
Intel's Orca DPO Pairs	11'573	1'286
Stanford Human Preferences	-	5'890

Table 1: Datasets used for DPO: The **M1** dataset, with 1522 unique exam questions and an average of 17.57 preference pairs per question, was processed in two ways. Firstly each preference pair is treated as an individual data point and secondly samples where both answers were under 200 characters were removed. The **WebGPT** dataset was processed by selecting the higher-scored answers as preferred and removing questions with tied scores. No processing was done to **ORCA** dataset. For the aforementioned datasets, a 9:1 train test split ratio was used. The **SHP** dataset processing is detailed in A.1.2.

2.3 Multiple choice questions

Anticipating Milestone 3, three relevant datasets suitable for fine-tuning the model on MCQs were collected. **Allenai's SciQ** dataset contains 13'679 science exam MCQs with a predefined train, evaluation and test split. The **Ai2 Arc Challenge** dataset contains 2'590 genuine grade-school level, science MCQs, with a predefined train, evaluation and test split. The **Measuring Massive Multitask Language Understanding** dataset (Hendrycks et al., 2021) contains 2'401 MCQs from 13 out of the 57 STEM topics and will be used only for testing purposes.

2.4 Performance evaluation

To assess model performance, a test dataset, named **Performance** dataset, was created using the test splits of the **DPO** and **stemQ** datasets and treating the chosen answer from the pair of answers as a golden answer. If a question appeared several times, the possible golden answers were stored in a list. This procedure reduced the dataset from 8'656 to 3'337 unique questions.

3 Model

3.1 Base Model

Various sequence-to-sequence (seq2seq) models including different Flan-T5s (Chung et al., 2022) (small 77M, base 248M and large 783M) and BARTs (Lewis et al., 2019) (base 140M and large 400M) were tested. Flan T5 is an enhanced version of the T5 (Raffel et al., 2023) encoder-decoder model that has been enriched through instruction fine-tuning across a broad spectrum of tasks and languages. BART is a seq2seq framework, featuring a bidirectional encoder and a left-to-right decoder.

3.2 SFT

Initially, Flan-T5 models underwent SFT to increase their performance in university-level science questions. The order of datasets during fine-tuning is important (Dodge et al., 2020). Two orders were compared: starting with stemQ data followed by EPFL course data and vice versa. The former order yielded slightly superior BLEU, ROUGE and BLEURT scores. Therefore, these models were chosen for further DPO training (A.2).

3.3 DPO

To enhance the alignment of the models with human preferences, the DPO training algorithm was applied using the TRL library (von Werra et al., 2024).

Low-rank Adaptation (LoRA): After extensive testing, it was found that the DPO algorithm necessitated an extended training period and exhibited inconsistency in reducing the training loss. Consequently, LoRA was chosen as a solution. LoRA is an efficient lightweight parameter efficient fine tuning that substantially reduces the number of trainable parameters during fine-tuning.

DPO Training: During the DPO training phase, the provided default sigmoid loss function was used. Additionally, LoRA was incorporated with specific parameters, including an r value set to 16, $alpha$ set to 32, and a dropout probability of 0.05. Furthermore, different configurations for the learning rate, batch size, and maximum sequence length were tested to optimize performance (Figure 1).

4 Preliminary Training Results

4.1 Evaluation metrics

As performance metrics, the Exact Match (EM), F1, BLEU, ROUGE and BLEURT were computed on the Performance dataset for different models (Figure 3). Test reward accuracies, corresponding to the mean of how often the chosen rewards are superior than the corresponding rejected rewards, were calculated at the end of each epoch and post training on DPO test datasets.

4.2 Hyper-parameter choices

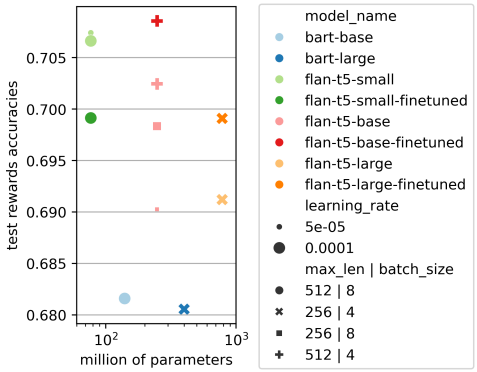

Figure 1: The test rewards accuracies obtained after the last training epoch for the different models and hyper-parameters. The test set used is the combination of the 10% test set splits of the M1, ORCA and WebGPT datasets.

Two different batch sizes, learning rates and maximum number of tokens were tested depending on the models. A limitation came from the trade-off between the model size and the available GPU RAM. Therefore, large models needed a smaller batch size and maximum tokens. Choosing BART as the base model led to less accurate reward predictions than the Flan-T5 models. Therefore, BART was removed from contention and not fine-tuned. Having an SFT stage prior to DPO seems to increase the accuracy by $\sim 0.5\%$ for both the base and the large Flan-T5s. However, it decreases the accuracy of the small model by the same magnitude (Figure 1). A learning rate of 10^{-4} led to better results when tested on the Flan-T5 base model. Therefore, most of the subsequent runs used this value. The best performing models in all regimes have very similar test rewards accuracies and only deviate by $\sim 0.5\%$. Therefore, using solely this metric, no clear choice could be made on the model size to choose.

4.3 Observations

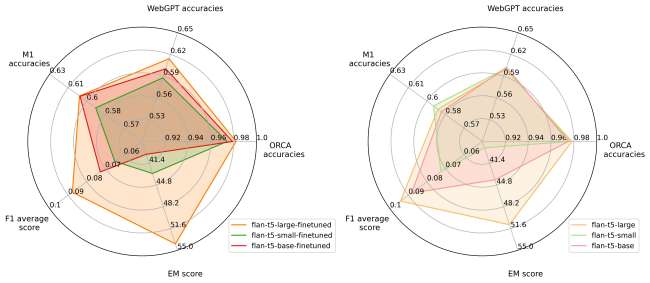

Figure 2: Evaluation of Flan-T5 small, base, and large performances at each model's best training epoch. Different metrics were used: the post-training policy reward accuracies on the ORCA, M1 and WebGPT datasets and the F1 average score and the EM score on the Performance dataset.

Depicted in Figure 2, all models exhibit very similar accuracy performances across the different test DPO datasets. The small difference in the average of reward accuracies between Figures 1 and 2 are probably due to inherent default parameters that change the behaviour of the model post-training. The **fine-tuned Flan-T5 large model** was finally chosen because it outperforms the other models in text generation quality, particularly in terms of the EM and F1 metrics. It also scores better on the M1 dataset than any other model.

5 Quantization Specialization

Different quantization methods will be explored to reduce the model size while maintaining performance. Initially, the BitsAndBytes library (Dettmers, 2024) will be used for 4-bit and 8-bit quantization. Experimentation with offloading, outlier threshold adjustments and nested quantization techniques will also be done. Additionally, linear quantization methods using the Quanto library will be explored.

References

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling Instruction-Finetuned Language Models. ArXiv:2210.11416 [cs].

Tim Dettmers. 2024. TimDettmers/bitsandbytes. Original-date: 2021-06-04T00:10:34Z.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping.

Iddo Drori, Sarah Zhang, Zad Chin, Reece Shuttleworth, Albert Lu, Linda Chen, Bereket Birbo, Michele He, Pedro Lantigua, Sunny Tran, Gregory Hunter, Bo Feng, Newman Cheng, Roman Wang, Yann Hicke, Saisamrit Surbehera, Arvind Raghavan, Alexander Siemenn, Nikhil Singh, Jayson Lynch, Avi Shporer, Nakul Verma, Tonio Buonassisi, and Armando Solar-Lezama. 2023. A dataset for learning university stem courses at scale and generating questions at a human level. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press.

Kavita Ganesan. 2018. Rouge 2.0: Updated and improved measures for evaluation of summarization tasks.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. ArXiv:1910.13461 [cs, stat].

Ismail Muraina. 2022. Ideal dataset splitting ratios in machine learning algorithms: General concerns for data scientists and data analysts.

Matt Post. 2018. A call for clarity in reporting bleu scores.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. ArXiv:1910.10683 [cs, stat].

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. Bleurt: Learning robust metrics for text generation.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, and Nathan Lambert. 2024. TRL: Transformer Reinforcement Learning. Original-date: 2020-03-27T10:54:55Z.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

A Appendix

A.1 Data

A.1.1 EPFL course dataset

The courses from which the questions were extracted are: Machine Learning (CS-433), General physics (PHYS-101(c)), Cryptography and Security (COM-401), Information security and privacy (COM-402) and Introduction to operating systems (CS-323).

A.1.2 Stanford Human Preferences Dataset (SHP)

This dataset contains 385'563 collective human preferences over responses to questions and instructions in 18 different subject areas from which only the four being related to university science questions were chosen. Each question has two answers with different scores. Based on these scores, the chosen and the rejected answer were determined. A total of 121'693 questions was built.

A.2 Fine-tuning

For the fine-tuning phase, two distinct datasets within the domain of university-level science questions were used. The aim was to fine-tune the model within this specific domain.

Each dataset was divided into training, evaluation and test subsets, with 80% allocated for training, and 10% each for evaluation and testing. This seemed to be a reasonable choice for the dataset and task in question (Muraina, 2022). Various combinations of arguments within Seq2SeqTrainingArguments, also impacting the Seq2SeqTrainer, were tested.

During training, a prompting strategy tailored for the Flan T5 model, appending 'Question:' at the beginning and 'Answer:' at the end of each question, was used. This prompt is proved to be effective for Flan T5 when general questions form the input (Wei et al.).

Initially, the Flan T5 small, large and base model underwent individual fine-tuning on either the EPFL course dataset or the stemQ dataset. Subsequently, the saved models were further fine-tuned with the remaining dataset.

The evaluation losses of the combined fine-tuned models starting with a Flan T5 base, even if only slightly, continue to decrease. Notably, the model fine-tuned first on the stemQ dataset and then on the EPFL course dataset has a higher loss. This can be explained by the fact that the EPFL course dataset has probably more complex and specific questions than the stemQ dataset and that the models are evaluated on their second fine-tuning dataset. Applying the same fine-tuning procedure on Flan T5 small and large provided similar results.

Evaluation metrics including BLEU (Post, 2018), ROUGE (Ganesan, 2018), and BLEURT (Sellam et al., 2020) scores were computed for both combined models and all Flan T5 sizes using a test dataset composed of 10% of the EPFL course dataset and 10% of the stemQ dataset. These evaluation metrics were better for the models, regardless of their size, when first fine-tuned on the stemQ dataset and subsequently on the EPFL course dataset. Therefore, for further steps, only the models fine-tuned in the better manner were considered.

Despite the fact that, in the end, the fine-tuning step did not result in a significant improvement in the final model performance, it remained a crucial component in the project pipeline's logical structure.

A.3 Model performance evaluation

The table on *Figure 3* compares the BLEURT score of the different trained Flan T5 models with the base models. The base models are the original models without any DPO or SFT training. For each of the 3'337 questions of the performance evaluation dataset, a BLEURT score was computed for both the trained and the base model. In the table, the percentages out of all 3'337 questions when the trained model attained a higher or a lower BLEURT score than the base model are shown. In case of equality, the question was not taken into account.

	Bleurt > base model	Bleurt < base model
Flan T5 small DPO	0.373089601	0.253221456
Flan T5 small SFT DPO	0.274198382	0.295175307
Flan T5 base DPO	0.443512137	0.250224753
Flan T5 base SFT DPO	0.352412346	0.320647288
Flan T5 large DPO	0.450704225	0.227449805
Flan T5 large SFT DPO	0.397362901	0.263110578

Figure 3: Table showing the percentages of questions where the trained model achieved a higher or lower BLEURT score than the corresponding base model. The data illustrates that across nearly all trained models, the BLEURT score surpasses that of the base model more frequently than it falls short. This trend strongly suggests the utility of the SFT and DPO trainings.

A.4 DPO Training

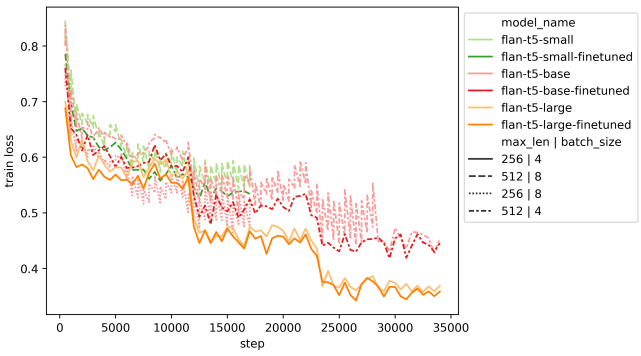

Figure 4: Train loss by training step. BART losses are omitted because they are not of the same magnitude.

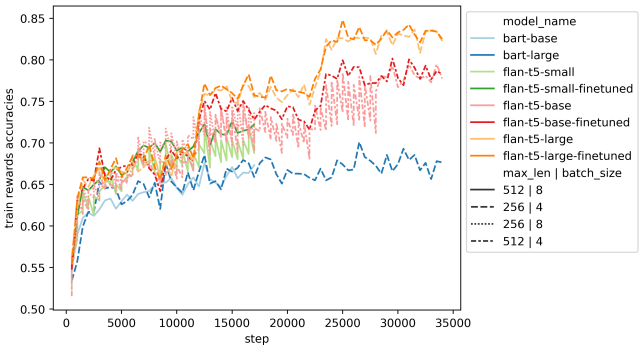

Figure 5: Train rewards accuracies by training step.

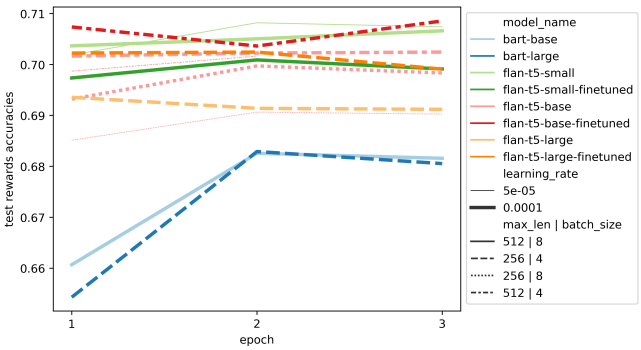

Figure 6: Test rewards accuracies by training epoch.