Enhancing Educational Dialogues: A Reinforcement Learning Approach for Generating AI Teacher Responses

Thomas Huber and **Christina Niklaus** and **Siegfried Handschuh**
University of St. Gallen
firstname.lastname@unisg.ch

Abstract

Reinforcement Learning remains an underutilized method of training and fine-tuning Language Models (LMs) despite recent successes. This paper presents a simple approach of fine-tuning a language model with Reinforcement Learning to achieve competitive performance on the BEA 2023 Shared Task whose goal is to automatically generate teacher responses in educational dialogues. We utilized the novel NLPO algorithm that masks out tokens during generation to direct the model towards generations that maximize a reward function. We show results for both the t5-base model with 220 million parameters from the HuggingFace repository submitted to the leaderboard that, despite its comparatively small size, has achieved a good performance on both test and dev set, as well as GPT-2 with 124 million parameters. The presented results show that despite maximizing only one of the metrics used in the evaluation as a reward function our model scores highly in the other metrics as well.

1 Introduction

Controlling the output of Language Models is a challenging problem in the field of Natural Language Processing (NLP). Recently Reinforcement Learning (RL) has successfully been applied to the training and fine-tuning of Language Models. ChatGPT, based on InstructGPT (Ouyang et al., 2022a), makes use of Reinforcement Learning. Ramamurthy et al. (2023) have proposed the GRUE (General Reinforced-language Understanding Evaluation) benchmark that consists of a variety of different tasks, supervised by different Reward Functions to measure the quality of the trained models. The reported results on a variety show good results on a variety of tasks. Despite recent advances in applying RL to the training and fine-tuning of LMs and their wide applicability to different tasks and benchmarks this approach is still not widely applied.

In this paper we make use of Reinforcement Learning-based fine-tuning to tackle the BEA 2023 Shared Task (Tack et al., 2023). The goal of the task is the generation of teacher-like responses in an educational dialogue setting between a student and a teacher. This necessitates that the language model can mimic the tone and overall quality of the teacher response. We have employed an approach that pushes the generations of the model in the right direction through the use of BERTScore as a reward function and using Reinforcement Learning as our training strategy.

Our model submission to the leaderboard is the implementation of the T5 model (Raffel et al., 2020) in the HuggingFace repository, t5-base with 220 million parameters. As the goal is to generate a response given an input dialogue we have chosen a sequence-to-sequence model. We follow the findings of Ramamurthy et al. (2023) who suggest that a small model with a high-quality reward function can match or outperform models with magnitudes of more parameters. For the training process we use the dialogue preceding the final teacher response as input and the final teacher response as the reference text. We achieve an average rank across all metrics of 5.38, out of 10 submissions, placing overall in seventh place on the leaderboard. For the DialogRPT maximum weighted ensemble metric our model achieves first place on the test set.
We additionally present results for an autoregressive model. The chosen model is the base GPT-2 model from the HuggingFace repository with 124 million parameters. The autoregressive model outperforms our submitted model despite its smaller size in terms of parameters, suggesting that this model architecture may be more suitable for this task.

2 Related Work

Ramamurthy et al. (2023) present results showing that Reinforcement Learning can be applied

Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023), pages 736–744
July 13, 2023

Tokenizer	Min	Max	Avg.
t5-base	201	9	99.17
gpt2	223	11	100.03

Table 1: Lengths of the training samples. Values are measured in tokens.

successfully in various NLP settings, including on the DailyDialog dataset (Li et al., 2017), which is similar in structure to the BEA task's dataset. Liu et al. (2021) present an approach to make language model generations less politically biased using Reinforcement Learning. Toledo et al. (2023) demonstrate the viability of a Reinforcement Learning approach in text-based games. Notably they achieve improvements over the previous state of the art in this zero-shot setting. The task of aiding students is comparable due to the large number of possible topics and unforeseen behavior of students when interacting with either a human teacher or a machine teacher. While it is not specifically considered in this task and underrepresented in current research, likely due to the current state of research in this area, there is the possible danger of models becoming outdated in the future, possibly very quickly, as the world around us changes. A solution for this is of course to re-train the models on new data to update them, but a strong performance in a zero-shot setting circumvents this problem altogether, and Reinforcement Learning approaches show viability in this area.

3 Data

The training data provided for the task by the organizers consists of 2747 samples of student-teacher dialogues from the Teacher Student Chatroom Corpus (Caines et al., 2020, 2022). There are always two speakers, a student and a teacher, and they take turns talking. Each of the samples contains one response. Each dialogue turn is prefixed with *teacher:* or *student:*, respectively. We use the full input dialogue as the input text, separating each speaker turn by newline. The reference text is the teacher response that follows the input dialogue. We used the t5-base model as well as the gpt2 model from HuggingFace and their respective tokenizers. Table 1 shows the lengths of the official training set released for the task.

To avoid potential issues or the need to cut off samples from the test set we have padded all the input tokens to a length of 256 tokens for our model. We note that the task description states that each passage is at most 100 tokens long. The difference in maximum lengths likely comes from our chosen tokenizers, which uses a different tokenizing strategy than the approach that was used to calculate the expected maximum length of 100 tokens. For the training process we used a 80/10/10 split for training-validation-testing of the released training data.

4 Approach

Below, we present the methods we developed to generate teacher responses in real-world samples of teacher-student interactions.

4.1 Reinforcement Learning in NLP

Our submission to the task leaderboard is a sequence-to-sequence-based model. The task is structured in a way that is suited for these kinds of models: Given an input sequence of student-teacher dialogues, the output is another sequence, the response of the teacher. The comparatively small size of the data set and simplicity of the data set allows fast prototyping and experimentation. One research area where problems are also often small is that of Reinforcement Learning (Sutton and Barto, 2018). While combining Reinforcement Learning with human feedback is an active field (Knox and Stone, 2008; Arumugam et al., 2019; Li et al., 2019; Christiano et al., 2023), it has only recently started being used in the field of NLP (Ziegler et al., 2019; Ouyang et al., 2022b; Lambert et al., 2022). Most importantly, the RL4LMs framework (Ramamurthy et al., 2023) has enabled the easy adaptation of RL approaches for NLP tasks. The authors have applied their framework to similar tasks, notably the IMDB review continuation, using the dataset by Maas et al. (2011). They achieved good results on this task using GPT2. They further report good results using T5 (Raffel et al., 2020) for a summarization task on news (Hermann et al., 2015) as well as the CommonGen task (Lin et al., 2019).

4.2 T5

In the spirit of research we have initially decided to use T5 for this task instead of following the findings of the authors and using GPT2 due to the task's similarity to the IMDB task. The compatibility of our chosen model with both being fine-tuned with

Reinforcement Learning as well as being usable in the RL4LMs framework has been demonstrated on a different task, so we conclude that our approach, while admittedly unusual, is not entirely unfounded in prior research.

4.3 GPT-2

Due to the relatively low ranking on the leaderboard of our T5 model we have additionally fine-tuned a GPT-2 checkpoint from the HuggingFace repository, with 124 million parameters, after the task concluded. As such this model was not submitted to the leaderboard. We include the configuration used for the training of both models in the appendix.

4.4 Algorithm

We follow the findings of Ramamurthy et al. (2023) and use their NLPO algorithm for the policy optimization during training. The performance of this algorithm is reported as the highest. It is an extension of the PPO algorithm (Schulman et al., 2017) and masks unlikely actions to reduce the action space. In the context of language generation this means masking next tokens whose cumulative probability is below a certain threshold. This reduction of the action space is important in the context of natural language problems as the action space in these contexts can be quite large. In the context of Reinforcement Learning a policy is a probability distribution over actions given a state. In our approach the policy is the language model being fine-tuned. The state is the generated tokens and the action is the next token to be generated in a language generation setting. Considering a language model itself to be a policy is a concept that has been used before in Liu et al. (2021) but is not widespread yet.

4.5 Reward Function

As our reward function we have chosen a pragmatic approach. We decided to use one of the metrics used in the evaluation as the reward function, as that should allow us to train the model to achieve a high score. The possibility of doing this showcases an advantage that a Reinforcement Learning-based approach has over other, more traditional approaches (both classic Machine Learning and Deep Learning) in the field of NLP: To lessen the gap between the evaluation criteria and the loss during training. Approaches for this problem exist (Song et al., 2016; Casas Manzanares et al., 2018) but it remains an open problem. This mismatch can be avoided by using Reinforcement Learning, and, in theory, should allow a high performance on a variety of tasks. Ramamurthy et al. (2023) report that the quality of the reward function has a greater effect on the performance of the model than the amount of training data. To keep our reward function clear we have opted to use only one metric as the reward signal, as opposed to combining all the evaluation metrics into one function that calculates a scalar value. We experimented with using the average of all the evaluation metrics as the reward but empirically found quickly that this does not yield good performance and have not pursued this direction further. The metrics for the BEA task are BERTScore (Zhang et al., 2020) and DialogRPT updown, human vs. rand and human vs. machine scores (Gao et al., 2020). We wanted to avoid the potential issue of reward hacking and thus decided not to use the updown score as a metric, as it seemed potentially prone to that issue. The other two DialogRPT scores were eliminated due producing very high scores (above 0.95) even early on during training and thus are unlikely to be useful as reward signals, as any improvements that the model learns could only lead to marginal increases in reward. For this reason we have chosen to use the BERTScore, specifically the F1, as our reward function.

5 Results

In Table 2 we present the outputs by a zero-shot t5-base model, our fine-tuned t5-base model and our fine-tuned GPT-2 model. Model output were not trimmed or modified. We note that the both the fine-tuned T5 and GPT-2 include prefixes in their responses in some cases. The GPT-2 model is especially prone to outputting a "student:" response, which is not the goal of the task. This does not have an overly negative effect on the evaluation metrics however. Further investigation of the alignment of the task metrics with the stated goal of generative models assuming the rule of teacher in student-teacher dialogues is recommended for this reason. Prompting the models by using the dialogue and adding a "teacher:" prompt at the end guided the models towards first writing a teacher response and only after that, on occasion, further student responses. To minimize assumptions and to modifying the task to improve our results we have not pursued the evaluation in this direction, and

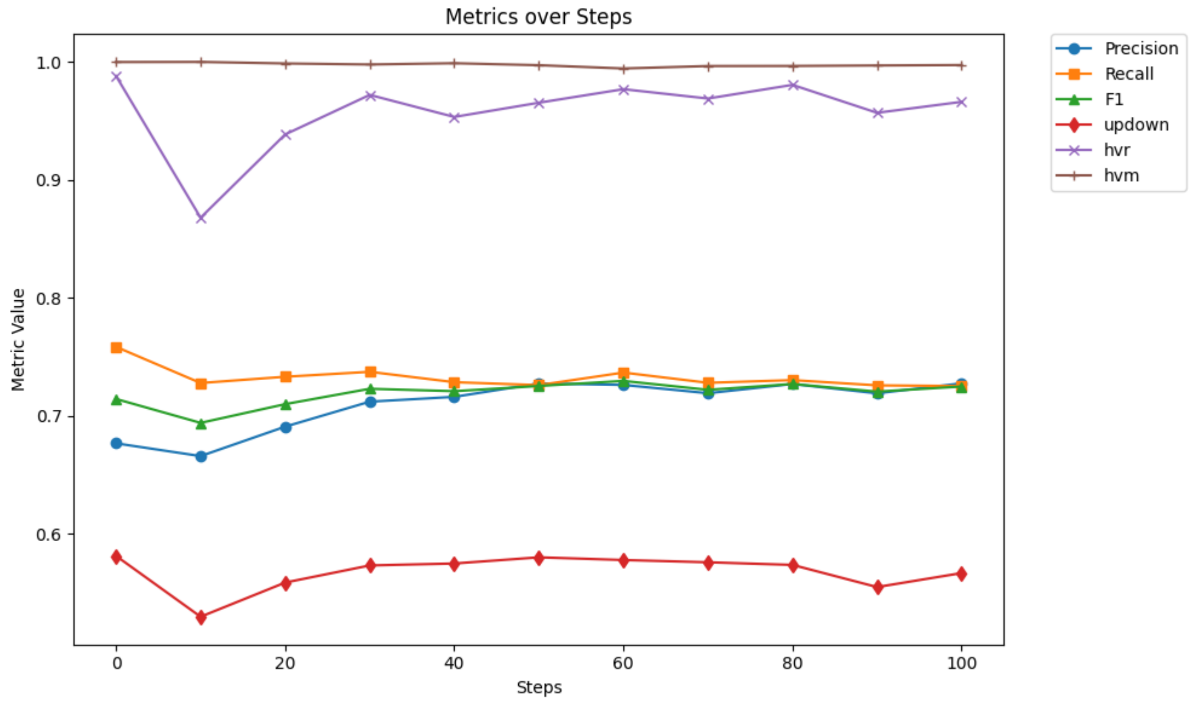

Figure 1: Metrics during the training process on the validation set for the GPT-2 model.

instead evaluated the models only on their output when given a dialogue, without any further prompting or modification.

5.1 Training Performance

Figure 1 shows the scores our GPT-2 model has achieved during the training process on the validation set. The scores of the trained model as well as zero-shot performance on the validation set are reported in Table 3. Due to an error the validation set splits were not pure during the training process of the T5 model and we do not include it in the graphic above.

5.2 Test Set Performance

We present the results of the evaluation on the test set in Table 4. Model outputs were generated on the test data dialogues, with the prefixes included, and were not pruned. Models often included wrong prefixes such as "student:" in their response. We did not remove these or filter the outputs for the first "teacher:" response. GPT-2 responses were set to have a minimum length of 12 and a maximum length of 100.

6 Conclusion

In this work we have shown our Reinforcement Learning-based approach on the BEA 2023 Shared Task. We have used a relatively simple approach and trained two models, t5-base with 220 million parameters and gpt2 with 124 million parameters. Despite the overall performance of the models being mixed we have achieved good results in some areas. The GPT-2 model has achieved a good performance on the task and is showing clear gains in terms of evaluation metrics over a zero-shot approach on the same data. This suggests that Reinforcement Learning-based fine-tuning of language models is a valid approach. According to previous work in the area the model performance when fine-tuned with Reinforcement Learning is strongly influenced by the quality of the reward function. Our approach to this task was very basic and leaves room for improvement, which we believe can be achieved by using both higher quality models instead of relatively small ones with few parameters as well as an improved reward function that makes use of multiple evaluation metrics.

ID	Zero-Shot t5-base	t5-base with RL	GPT-2 with RL	Correct Labels
test_0001	student: 'extreme weather' and 'extreme climate'!	yes!	student: Yes, that's right.\n student: Solar/wind energy is a great idea	yes fantastic - so one answer is to 'reduce carbon emissions' (Collocation!) - what kind of energy is e.g. solar/wind
test_0002	Maybe some of them were too young!	To please the researchers?	teacher:\n teacher.\n teacher.\n	Could you make a sentence with a modal? Some of them ___ ___ ___ too young
test_0003	student: or - my train leaving time at 9:30.	ok?	student:...\n student:...\n student:...\n'	So: 'I'm going home tomorrow - my train ____ at 9:30' One word...what is it?
test_0004	so my time was faster/quicker/better than yesterday!	OK!	teacher: ok... so my time was faster.\n teacher:	OK so one more: My personal best for 5k is 15:03 (along time ago) ..... So 15:03 is the ____ I ____ ___ the 5k.??
test_0005	student: right?	teacher:..does it matter?	student: yes, that's right.\n student:...	OK see you in 15 mins ok?

Table 2: Comparison of responses from trained model, untrained model, and the correct labels, for our T5 model.

	BERTScore			DialogRPT (Avg/Max)		
Model	**Precision**	**Recall**	**F1**	**updown**	**hvr**	**hvm**
`GPT-2 (zero-shot)`	0.65	0.69	0.67	0.65/0.84	0.99/1.0	1.0/1.0
`GPT-2 (RL)`	0.73	0.72	0.72	0.57/0.80	0.97/1.0	0.90/1.0

Table 3: Evaluation metrics for the fine-tuned GPT-2 model and zero-shot performance of the untrained model on the validation set.

	BERTScore			DialogRPT (Avg/Max)		
Model	**Precision**	**Recall**	**F1**	**updown**	**hvr**	**hvm**
`T5 (zero-shot)`	0.71	0.69	0.70	0.62/0.85	0.98/1.0	0.95/1.0
`T5 (RL, submitted)`	0.76	0.65	0.70	0.50/0.70	0.92/1.0	0.88/1.0
`GPT-2 (zero-shot)`	0.68	0.65	0.66	0.67/0.85	1.0/1.0	0.99/1.0
`GPT-2 (RL)`	0.77	0.66	0.71	0.59/0.80	0.98/1.0	0.96/1.0

Table 4: Evaluation metrics on the official test set. Scores were calculated using the released labels. Model inputs included the speaker prefix. Outputs were not pruned or filtered and often included a prefix.

References

Dilip Arumugam, Jun Ki Lee, Sophie Saskin, and Michael L Littman. 2019. Deep reinforcement learning from policy-dependent human feedback. *arXiv preprint arXiv:1902.04257*.

Andrew Caines, Helen Yannakoudakis, Helen Allen, Pascual Pérez-Paredes, Bill Byrne, and Paula Buttery. 2022. The teacher-student chatroom corpus version 2: more lessons, new annotation, automatic detection of sequence shifts. In *Swedish Language Technology Conference and NLP4CALL*, pages 23–35.

Andrew Caines, Helen Yannakoudakis, Helena Edmondson, Helen Allen, Pascual Pérez-Paredes, Bill Byrne, and Paula Buttery. 2020. The teacher-student chatroom corpus. *arXiv preprint arXiv:2011.07109*.

Noé Casas Manzanares, José Adrián Rodríguez Fonollosa, and Marta Ruiz Costa-Jussà. 2018. A differentiable bleu loss. analysis and first results. In *ICLR 2018 Workshop Track: 6th International Conference on Learning Representations: Vancouver Convention Center, Vancouver, BC, Canada: April 30-May 3, 2018*.

Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2023. Deep reinforcement learning from human preferences.

Xiang Gao, Yizhe Zhang, Michel Galley, Chris Brockett, and Bill Dolan. 2020. Dialogue response ranking training with large-scale human feedback data.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28.

W Bradley Knox and Peter Stone. 2008. Tamer: Training an agent manually via evaluative reinforcement. In *2008 7th IEEE international conference on development and learning*, pages 292–297. IEEE.

Nathan Lambert, Louis Castricato, Leandro von Werra, and Alex Havrilla. 2022. Illustrating reinforcement learning from human feedback (rlhf). *Hugging Face Blog*. Https://huggingface.co/blog/rlhf.

Guangliang Li, Randy Gomez, Keisuke Nakamura, and Bo He. 2019. Human-centered reinforcement learning: A survey. *IEEE Transactions on Human-Machine Systems*, 49(4):337–349.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. DailyDialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 986–995, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2019. Commongen: A constrained text generation challenge for generative commonsense reasoning. *arXiv preprint arXiv:1911.03705*.

Ruibo Liu, Chenyan Jia, Jason Wei, Guangxuan Xu, Lili Wang, and Soroush Vosoughi. 2021. Mitigating political bias in language models through reinforced calibration.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022a. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Siqi Ouyang, Rong Ye, and Lei Li. 2022b. On the impact of noises in crowd-sourced data for speech translation. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 92–97, Dublin, Ireland (in-person and online). Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Rajkumar Ramamurthy, Prithviraj Ammanabrolu, Kianté Brantley, Jack Hessel, Rafet Sifa, Christian Bauckhage, Hannaneh Hajishirzi, and Yejin Choi. 2023. Is reinforcement learning (not) for natural language processing: Benchmarks, baselines, and building blocks for natural language policy optimization.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms.

Yang Song, Alexander G. Schwing, Richard S. Zemel, and Raquel Urtasun. 2016. Training deep neural networks via direct loss minimization.

Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

Anaïs Tack, Ekaterina Kochmar, Zheng Yuan, Serge Bibauw, and Chris Piech. 2023. The BEA 2023 Shared Task on Generating AI Teacher Responses in Educational Dialogues. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications*, page to appear, Toronto, Canada. Association for Computational Linguistics.

Edan Toledo, Jan Buys, and Jonathan Shock. 2023. Policy-based reinforcement learning for generalisation in interactive text-based environments. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1230–1242, Dubrovnik, Croatia. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

A Appendix

We include our RL4LMs configuratiosn used for training. The configuration seen in Figure 2 shows the configuration for the submitted T5 model. The reward function `bertscore_bea` is the F1 BERTScore, using the "distilbert-base-uncased" model, with the prefixes removed before the rewards are calculated. Figure 3 shows the configuration for the GPT-2 model. The reward function does not remove the prefixes before calculating the reward.

```yaml
tokenizer:
  model_name: t5-base
  padding_side: left
  truncation_side: left
  pad_token_as_eos_token: False

reward_fn:
  id: bertscore_bea
  args:
    language: en

datapool:
  id: bea_full_seq2seq_splits_onlyResponse
  args:
    file_path: "/data/bea/data/release_1_train_dev/train_with-reference.jsonl"

env:
  n_envs: 1
  args:
    max_prompt_length: 256
    max_episode_length: 100
    terminate_on_eos: True
    prompt_truncation_side: "right"
    context_start_token: 0

alg:
  id: nlpo
  args:
    n_steps: 128
    batch_size: 64
    verbose: 1
    learning_rate: 0.00001
    n_epochs: 5
    ent_coef: 0.0
    gae_lambda: 0.9
    vf_coef: 0.1
  kl_div:
    coeff: 0.02
    target_kl: 2
  policy:
    id: maskable_seq2seq_lm_actor_critic_policy
    args:
      model_name: t5-base
      apply_model_parallel: True
      mask_type: "learned_top_p"
      top_mask: 0.9
      target_update_iterations: 20
      generation_kwargs:
        do_sample: True
        min_length: 20
        top_k: 200
        max_new_tokens: 100  # this must align with env’s max steps

train_evaluation:
  eval_batch_size: 100
  n_iters: 100
  eval_every: 10
  save_every: 10
  metrics:
    - id: bertscore_bea
      args:
        language: en
    - id: bert_score
      args:
        language: en
```

Figure 2: RL4LMs configuration used for training the T5 model.

```yaml
tokenizer:
  model_name: gpt2
  padding_side: left
  truncation_side: left
  pad_token_as_eos_token: True

reward_fn:
  id: bertscore_bea_distil
  args:
    language: en

datapool:
  id: bea_full_seq2seq_splits_onlyResponseNoShuffle
  args:
    file_path: "/data/bea/data/release_1_train_dev/train_with-reference.jsonl"


env:
  n_envs: 1
  args:
    max_prompt_length: 256
    max_episode_length: 100
    terminate_on_eos: True

alg:
  id: nlpo
  args:
    n_steps: 128
    batch_size: 64
    verbose: 1
    learning_rate: 0.00001
    n_epochs: 5

  kl_div:
    coeff: 0.1
    target_kl: 1.0
  policy:
    id: maskable_causal_lm_actor_critic_policy
    args:
      model_name: gpt2
      apply_model_parallel: True
      top_mask: 0.9
      min_tokens_to_keep: 100
      mask_type: 'learned_top_p'
      target_update_iterations: 5
      generation_kwargs:
        do_sample: True
        min_length: 12
        max_new_tokens: 100

train_evaluation:
  eval_batch_size: 100
  n_iters: 100
  eval_every: 10
  save_every: 10
  metrics:
    - id: bertscore_bea_distil
      args:
        language: en
```


Figure 3: RL4LMs configuration used for training the GPT-2 model.