Reinforcement Learning-Enhanced Abstractive Summarization: Optimizing Seq2Seq Models Beyond Supervised Learning Constraints

Arjyahi Bhattacharya[1], Charith Purushotham[1]
[1]Department of Computer Science, University of Colorado Boulder

Abstract

Our project studies the application of reinforcement learning (RL) techniques, Self-Critical Sequence Training (SCST) and Proximal Policy Optimization (PPO), to boost text summarization performance on the SAMSum dataset. We compare the effectiveness of these RL fine-tuning methods against a baseline sequence-to-sequence model using the BLEU score as the primary evaluation metric. Our experiments demonstrate that both SCST and PPO significantly improve summarization quality, with PPO (batch size 8) achieving the highest BLEU score of 0.9246, surpassing the base model by 5.42%. These findings highlight the effectiveness of RL-based fine-tuning in producing more coherent and human-like summaries. We also discuss the role of batch size and suggest future directions, which include generalization to other domains, and integration with pretrained transformer models.

1 Introduction

With the rapid increase in digital content there is a lot of information out there, which creates a need for tools that can effectively summarize large amounts of text. Text summarization plays an important role across many domains, such as news, customer service, and science, where quick understanding of large volumes of text is highly desired. Among available methods, extractive summarization, i.e., pulling key sentences directly from source text, often outputs in garbled results. Abstractive summarization, however, rephrases content naturally, making it more desirable but also more challenging.

Most state-of-the-art abstractive summarization models are trained using supervised learning, where token-level losses like cross-entropy are optimized. However, these objectives often poorly align with the metrics used for evaluating summary quality, such as BLEU or ROUGE. This mismatch can lead to outputs that are grammatically correct but semantically weak or uninformative. Reinforcement learning (RL) offers a promising solution by enabling direct optimization for these end-goal evaluation metrics. While RL has been explored in large transformer-based models, its impact on smaller, resource-efficient architectures remains underexplored.

In this project, we focus on the task of single-turn abstractive dialogue summarization using the SAMSum dataset**.** Each input consists of a multi-turn dialogue involving two or more speakers, and the goal is to produce a concise, readable summary that accurately reflects the key points of the conversation. As a baseline, we employ a GRU-based sequence-to-sequence model. To improve its performance, we explore two reinforcement learning-based fine-tuning methods: Self-Critical Sequence Training (SCST) and Proximal Policy Optimization (PPO)[8]. We evaluate model performance using the BLEU score, which assesses the degree of n-gram overlap between the generated summaries and human-written references. The research question we address is: *Can RL-based fine-tuning (via SCST and PPO) improve the quality of abstractive dialogue summaries produced by a GRU-based model, as measured by BLEU scores?* This work aims to explore how RL can enhance summary quality beyond what is achieved by standard supervised baselines on a constrained, well-defined task.

2 Related Work

Paulus et al. (2017)[1]: This work introduced a model that combines supervised learning with RL, specifically using the REINFORCE algorithm, to optimize ROUGE scores. Their approach also incorporated intra-attention mechanisms to

reduce repetition in generated summaries. Building upon this, our work explores alternative RL algorithms, SCST and PPO, and evaluates their effectiveness using BLEU scores, providing a different perspective on optimizing summarization quality.

Keneshloo et al. (2019)[2]: Keneshloo et al. addressed the challenge of generalizing summarization models to new datasets by proposing a self-critical policy gradient approach within a transfer learning framework. Their method demonstrated improved generalization across various datasets. In contrast, our research focuses on enhancing summary quality within a specific dataset (SAMSum) by fine-tuning models using RL techniques, without explicitly targeting cross-domain generalization.

Farooq (2025)[3]: Farooq introduces a T5-based summarization model with hierarchical RL to adapt summary lengths based on time constraints. They compare PPO, A2C, and SAC on metrics like ROUGE and BERTScore. While their approach targets adaptive summary lengths and efficiency, our focus is on improving summary quality through RL fine-tuning with different reward signals, without time-based constraints.

Pulari et al. (2025)[4]: They propose using RL with human feedback and prompting techniques to enhance news summarization. Their approach introduces a new evaluation metric, H-Rouge, and emphasizes human-guided training improvements. Unlike their work, which uses human feedback for training, our study focuses on automatic rewards (e.g., BLEU) and explores a range of RL-based fine-tuning methods to improve summarization performance.

3 Methodology

3.1 Dataset and Preprocessing

We used the SAMSum Dataset (~16,000 dialogue-summary pairs from Kaggle), comprising real-life conversations and human-written summaries, ideal for training and evaluating abstractive summarization models.

Data Cleaning: We removed rows with missing values or duplicate dialogues using a custom DatasetCleaner class for consistent preprocessing.

Text Normalization: Informal chat-style data was standardized by:

- Lowercasing all text
- Removing HTML tags and URLs
- Expanding contractions and chat abbreviations (e.g., “I’m” - “I am”)
- Removing emojis
- Inserting <start> and <end> tokens in summaries to aid sequence modeling

Tokenization & Padding: A custom tokenizer trained on both dialogues and summaries mapped text to integer indices, using a dedicated <OOV> token for unknown words. Sequences were post-padded to fixed lengths based on maximum observed dialogue/summary lengths, ensuring uniform input/output shapes for batch training.

3.2 Model Development

Model architecture: Our approach uses an encoder-decoder framework enhanced with an attention mechanism:

- Encoder: A two-layer GRU processes embedded dialogue tokens to produce contextual hidden states.
- Decoder: A two-layer GRU generates the summary, using the encoder’s context and its internal state at each step.
- Attention: A multi-head attention mechanism helps the decoder focus on semantically relevant encoder outputs during generation.
- Output Layer: A fully connected linear layer with softmax produces token probabilities over the vocabulary.

Training setup: The model is trained using cross-entropy loss, appropriate for multi-class classification at each time step of the sequence. The Adam optimizer is employed with an initial learning rate of 0.001 for adaptive learning during training. Training is conducted with batch sizes of 8 and 16 for experimental purposes. The model was trained for 5 epochs, with both training and validation loss monitored. Training was performed using PyTorch with support for multi-GPU acceleration via DataParallel. To prevent overfitting and enhance generalization, an early stopping mechanism was implemented. If validation loss did not improve beyond a

minimum delta of 0.001 over 3 consecutive epochs, training was halted early.

3.3 Reinforcement Learning Fine-Tuning

To improve summary quality beyond cross-entropy optimization, we fine-tuned the pretrained model using SCST and PPO, with ROUGE-L as the reward signal to promote structural alignment with reference summaries. Summaries were generated using greedy decoding. In SCST, the baseline reward was derived from the model's own greedy output. PPO was initialized with the pretrained model and updated using clipped reward-weighted feedback for stable learning. In both strategies, cross-entropy loss was retained to preserve fluency alongside informativeness. Both RL methods were tested with batch sizes 8 and 16 to evaluate stability and reward learning efficiency.

3.4 Evaluation Setup

We evaluated all models using the SAMSum validation split for consistency. While ROUGE-L was used as the reward during RL, BLEU score served as the final evaluation metric due to its focus on n-gram precision, offering complementary insights into generation quality. We compared BLEU scores across:

- The baseline GRU-attention model
- SCST-enhanced model
- PPO-enhanced model

Each RL variant was evaluated at both batch sizes, with the best-performing configuration reported. Training stability was assessed via reward trends and validation loss across epochs. Our goal was to observe increased BLEU scores after finetuning the model using SCST and PPO.

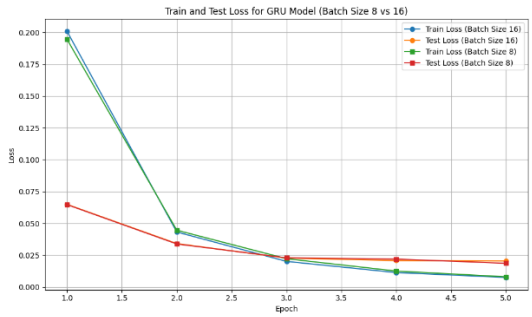
4 Experiments

The goal of our experiments is to evaluate the effectiveness of RL fine-tuning techniques, SCST and PPO, in improving a base model. All models were evaluated using the BLEU score on the SAMSum dataset, a benchmark for abstractive summarization of dialogue.

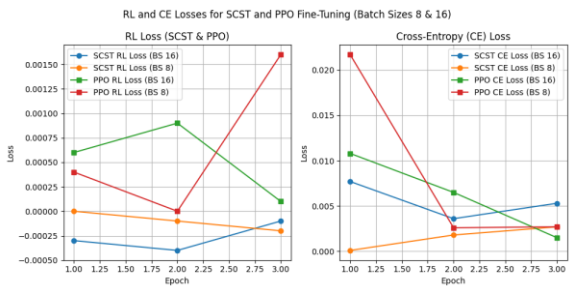
4.1 Experimental Setup

- Base model: A GRU-based sequence-to-sequence model with attention, trained with cross-entropy loss.
- RL fine-tuning methods:
    - SCST: A policy gradient method that uses a baseline (greedy-decoded summary) to reduce variance in reward optimization, with ROUGE-L as the reward function.
    - PPO: A more stable policy optimization method using a clipped surrogate objective.
- Batch sizes: Fine-tuning was conducted with batch sizes of 8 and 16 to assess performance sensitivity.
- Evaluation metric: BLEU score was used to assess final summary quality on the validation set, as it captures n-gram overlap between generated and reference summaries. Summaries were generated via greedy decoding for consistency.

4.2 Results



From the above graph, where the GRU models are trained only using supervised constraints, the model with batch size 8 demonstrates slightly better generalization, achieving a lower final test loss (0.0186) compared to the batch size 16 model (0.0204), though both converge effectively.



From the graphs above, we can see that PPO fine-tuning, especially with batch size 8, led to the steepest drop in CE loss early on. SCST fine-tuning showed more stable but smaller improvements, with batch size 8 converging more

smoothly. Overall, PPO yielded more aggressive CE loss reduction, while SCST offered steadier optimization.

Expt	Batch Size	Base BLEU	RL BLEU	Improvement
SCST	8	0.8800	0.9190	+4.43%
SCST	16	0.8382	0.8981	+7.14%
PPO	8	0.8771	0.9246	+5.42%
PPO	16	0.8540	0.9024	+5.66%

Both SCST and PPO led to significant performance gains. The PPO model with batch size 8 achieved the highest BLEU score of 0.9246, while SCST with batch size 16 showed the greatest relative improvement over its base. However, models fine-tuned with batch size 8 consistently outperformed those trained with batch size 16, suggesting smaller batch sizes yield better generalization in this context.

4.3 Findings

- RL fine-tuning consistently improved summarization performance across all configurations.
- PPO (batch size 8) yielded the best absolute BLEU score, while SCST (batch size 16) had the highest relative gain.
- Smaller batch sizes were more effective, emphasizing the importance of tuning RL hyperparameters like batch size.

4.4 Limitations:

- RL-based fine-tuning increases training time and computational overhead.
- Model performance was sensitive to batch size, indicating a need for careful tuning.
- The study is limited to the SAMSum dataset, further evaluation on diverse datasets is required to confirm generalizability.
- Overfitting risk may be higher with smaller batches, additional validation on held-out test sets is recommended.

5 Conclusion

This study evaluated the impact of reinforcement learning (RL) methods, Self-Critical Sequence Training (SCST) and Proximal Policy Optimization (PPO), on improving the performance of a base text summarization model using the SAMSum dataset. The experimental results clearly show us that both RL approaches notably boost summarization quality, as indicated in BLEU score improvements over the baseline. Our findings acknowledge the effectiveness of using reinforcement learning in guiding the summarization model toward generating more coherent and human-like outputs.

The experiments also revealed that smaller batch sizes tend to yield better performance, potentially due to improved generalization. Overall, this work demonstrates the value of reinforcement learning in fine-tuning summarization models and opens up new directions for enhancing text generation tasks.

Future Work:

While SCST and PPO improved summarization quality on the SAMSum dataset, several directions remain for further exploration:

- Generalization across tasks and domains: Applying these RL techniques to extractive summarization and other datasets can help evaluate their robustness and adaptability.
- Training efficiency: Given the computational cost of PPO, future work could explore faster methods like actor-critic models, curriculum learning, or multi-agent RL.
- Evaluation: To depict summary quality in a better way, future studies would include ROUGE, METEOR, and human evaluations alongside BLEU.
- Using pretrained models: Combining RL-based fine-tuning with pretrained transformers like T5 or BART could boost performance by integrating general language understanding with task-specific optimization.

In conclusion, this study demonstrates that reinforcement learning, particularly SCST and PPO, offers a promising path for advancing text summarization. With further exploration and optimization, these techniques can significantly contribute to more effective and adaptable summarization systems.

References

1. Paulus, R., Xiong, C., & Socher, R. (2017). A deep reinforced model for abstractive summarization. *Proceedings of the 2019 SIAM International Conference on Data Mining*. https://arxiv.org/pdf/1705.04304.

2. Keneshloo, Y., Ramakrishnan, N., & Reddy, C. K. (2017). Deep transfer reinforcement learning for text summarization. *Proceedings of the 2019 SIAM International Conference on Data Mining*, 675–683. https://doi.org/10.1137/1.9781611975673.76.

3. Farooq, A. (2025). Hierarchical reinforcement learning for adaptive text summarization. *Preprints*. https://www.preprints.org/manuscript/202503.2300

4. Pulari, S. R., Maramreddy, U., & Vasudevan, S. K. (2025). Improved fine-tuned reinforcement learning from human feedback using prompting methods for news summarization. *International Journal of Interactive Multimedia & Artificial Intelligence*. https://doi.org/10.9781/ijimai.2025.02.001

5. Gu, J., Cho, K., & Li, V. O. K. (2017). Trainable greedy decoding for neural machine translation. *arXiv:1702.02429v1 [cs.CL]*. https://arxiv.org/pdf/1702.02429.

6. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). *Proximal Policy Optimization Algorithms*. arXiv:1707.06347 [cs.LG]. https://arxiv.org/pdf/1707.06347

7. Wiseman, S., & Rush, A. M. (2016). Sequence-to-sequence learning as beam-search optimization. *arXiv:1606.02960v2 [cs.CL]*. https://arxiv.org/pdf/1606.02960.

8. Reddy, K. L., Shanmukh, M. P., Kumar, C., Kumar, T., Kumar, A., Kumar, P., & Venkatraman, K. (2024). *Enhancing abstractive text summarization with proximal policy optimization*. In *Proceedings of the 2024 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)* IEEE. DOI:10.1109/ICAECT60202.2024.10469299

9. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Lillicrap, T. P., Ghahramani, Z., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540), 529–533. https://doi.org/10.1038/nature14236

10. Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement learning: A survey. Journal of Artificial Intelligence Research, 4, 237–285. https://www.jair.org/index.php/jair/article/view/10166

11. Saadany, H., & Orăsan, C. (2021). *BLEU, METEOR, BERTScore: Evaluation of metrics performance in assessing critical translation errors in sentiment-oriented text*. arXiv:2109.14250 [cs.CL]. https://arxiv.org/pdf/2109.14250

12. Peters, J., & Schaal, S. (2007). Reinforcement learning by reward-weighted regression for operational space control. *ICML '07: Proceedings of the 24th International Conference on Machine Learning*, 745–750. https://doi.org/10.1145/1273496.1273590.

13. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. *MIT Press*.

14. Kaiser, Ł., & Bengio, Y. (2016). Learning to summarize with human feedback. *Proceedings of NeurIPS 2016*, 29. https://arxiv.org/abs/1606.05903.

15. Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *NeurIPS 2014*, 27, 3104–3112. https://arxiv.org/abs/1409.3215.

16. Yao, K., Zhang, L., Luo, T., & Wu, Y.** (2018). Deep reinforcement learning for extractive document summarization. *Neurocomputing*, 275, 1673–1682. https://doi.org/10.1016/j.neucom.2018.01.020.

17. Du, Y., Li, Z., Cheng, P., Chen, Z., Xie, Y., Wan, X., & Gao, A. (2025). Simplify RLHF as Reward-Weighted SFT: A Variational Method. *arXiv preprint arXiv:2502.11026*. https://arxiv.org/pdf/2502.11026.