Hierarchical Reinforcement Learning for Adaptive Text Summarization

Ahmad Farooq *

Posted Date: 31 March 2025

doi: 10.20944/preprints202503.2300.v1

Keywords: hierarchical reinforcement learning; adaptive text summarization; PPO; A2C; SAC; T5 model; *ROUGE*; *BERTScore*

Article

Hierarchical Reinforcement Learning for Adaptive Text Summarization

Ahmad Farooq

University of Arkansas at Little Rock; afarooq@ualr.edu

Abstract: This study presents a novel approach to adaptive text summarization using hierarchical reinforcement learning. We develop a T5-based hierarchical summarizer with a level selector, implementing and comparing three reinforcement learning algorithms: Proximal Policy Optimization (PPO), Advantage Actor-Critic (A2C), and Soft Actor-Critic (SAC). Our system adapts summary length based on time constraints and is evaluated using *ROUGE* and *BERTScore* metrics. Experiments conducted on the CNN/DailyMail dataset illustrate the potential of this approach in balancing summary quality and generation speed. Results show that PPO achieves the highest *ROUGE* and *BERTScore*s, while A2C demonstrates a better balance between quality and efficiency. The paper emphasizes the potential as well as challenges of employing reinforcement learning for adaptive summarization, paving the way for future research in this critical domain of natural language processing.

Keywords: hierarchical reinforcement learning; adaptive text summarization; PPO; A2C; SAC; T5 model; *ROUGE*; *BERTScore*

1. Introduction

The rapid expansion of textual information in the digital era has necessitated the development of efficient and adaptive text summarization systems. With the proliferation of digital content, particularly in areas like news, research, and social media, the ability to quickly distill key information from large documents or collections of texts has become increasingly crucial. This study presents a novel approach to text summarization using hierarchical reinforcement learning techniques, with the objective of developing a system that can adapt to different summarization needs while maintaining high-quality output.

Text summarization, the process of distilling a source text into a more concise form while maintaining its essential information, has been a long-standing challenge in natural language processing (NLP). Traditional approaches to summarization can be broadly categorized into two types: extractive and abstractive. Extractive methods select and present the most important sentences from the original text, while abstractive methods generate new sentences that capture the essence of the content [1]. While extractive methods have been widely used due to their simplicity, recent advances in neural network architectures and natural language generation have made abstractive summarization increasingly viable and effective [2].

However, existing summarization systems often face several limitations. Many are designed to produce summaries of fixed lengths, lacking the flexibility to adapt to different user needs or time constraints. Additionally, the quality of summaries can vary significantly depending on the length, complexity, and domain of the source text. Furthermore, the evaluation of summary quality remains a challenge, with traditional metrics like Recall- Oriented Understudy for Gisting Evaluation (*ROUGE*) [3] not always aligning with human judgments of quality, especially for abstractive summaries.

This project addresses these challenges by developing a hierarchical reinforcement learning approach for adaptive text summarization. Our system employs a T5-based hierarchical

 doi:10.20944/preprints202503.2300.v1

summarizer with a level selector, enabling it to generate summaries of varying lengths based on time constraints. We implement and compare three state-of-the-art reinforcement learning algorithms: Proximal Policy Optimization (PPO), Advantage Actor- Critic (A2C), and Soft Actor-Critic (SAC). By optimizing for both summary quality and generation speed, this method seeks to develop a flexible summarization model that can adapt to various summarization requirements. The primary objectives of this research are:

1. To develop a hierarchical reinforcement learning framework for text summarization that can adapt to different time constraints and summary length requirements.
2. To implement and compare the effectiveness of PPO, A2C, and SAC algorithms in the context of text summarization, offering insights into their relative strengths and weaknesses for this task.
3. To create a summarization system that balances summary quality and generation speed, addressing the practical need for efficient summarization in time-sensitive applications.
4. To evaluate the performance of the proposed system using a comprehensive set of metrics, including *ROUGE* scores and *BERTScore*, providing a comprehensive assessment of summary quality.
5. To analyze the impact of various reinforcement learning algorithms on the summa- rization process, including their ability to handle trade-offs between summary length, quality, and generation time.

To achieve these objectives, we utilize the CNN/DailyMail dataset [4], a widely used benchmark in text summarization research. This dataset contains a wide range of news stories and their related human-written summaries, providing a challenging and realistic test bed for our summarization system.

Our approach builds upon recent advancements in both summarization and rein- forcement learning. In the field of summarization, transformer-based models like BART [5] and T5 [6] have shown remarkable performance in generating high-quality abstractive summaries. By integrating these advanced language models into a reinforcement learning framework, we intend to develop a system capable of generating summaries that are both informative and coherent, while also being adaptable to different constraints.

The use of reinforcement learning in text summarization is a growing area of research [7]. Reinforcement learning enables the summarization model to learn from its own actions and optimize for long-term rewards, potentially leading to better alignment with human preferences compared to traditional supervised learning approaches. By exploring multiple reinforcement learning algorithms, we aim to offer insights into their relative effectiveness for this task and contribute to the broader understanding of reinforcement learning applications in natural language processing.

Our goal is to enhance adaptive text summarization systems through this study, of- fering a flexible and effective tool for navigating the ever-expanding world of digital information. By integrating the power of state-of-the-art language models with the adapt- ability of reinforcement learning, our work has the potential to impact a wide range of applications, from personalized news aggregation to automated research assistants and beyond.

2. Background and Related Work

2.1. Text Summarization

Text summarization has been an active area of research in the field of natural language processing for decades. The aim of text summarization is to condense a source text into a shorter version while retaining its key information and meaning. Summarization techniques can be broadly categorized into two main approaches: extractive and abstractive [8]. Extractive summarization involves selecting and presenting the most important sentences or phrases from the original text. Early approaches to extractive summarization relied on statistical methods,

 doi:10.20944/preprints202503.2300.v1

such as term frequency-inverse document frequency (*TF- IDF*) [9,10], to identify the most salient sentences. More recent methods have employed machine learning techniques, including supervised learning [11] and graph-based meth- ods [12], to improve the selection of relevant sentences.

On the other hand, abstractive summarization aims to generate new sentences that capture the essence of the original text. This approach is more challenging as it requires the system to understand the content, paraphrase, and potentially generate new phrases not present in the source text. With the advent of deep learning, particularly sequence-to- sequence models [13] and transformer architectures [14], abstractive summarization has seen significant advancements in recent years.

2.2. Neural Approaches to Summarization

The introduction of neural network-based models has revolutionized the field of text summarization. Sequence-to-sequence models with attention mechanisms [15] were among the first neural architectures to achieve notable success in abstractive text summariza- tion. These models can capture long-range dependencies in the text and generate more coherent summaries as compared to traditional methods. The development of transformer models [14] marked another significant milestone in summarization research. Models like `BERT` [16], `GPT` [17], and their variants have been fine-tuned for summarization tasks with impressive results. Notably, `BART` [5] and `T5` [6], which are based on the transformer architecture, have achieved state-of-the-art performance on various summarization bench- marks. Nonetheless, these models often struggle with generating summaries of varying lengths and adapting to different time constraints, which are essential requirements in numerous real-world applications. This has motivated research into more flexible and adaptive summarization approaches.

2.3. Reinforcement Learning in NLP

Reinforcement learning (RL) has emerged as a promising approach to address some of the limitations of supervised learning in NLP tasks, including summarization. RL enables models to learn from their own actions and optimize for long-term rewards, potentially leading to better alignment with human preferences. In the context of summarization, RL has been applied to optimize for various objectives beyond simple overlap with reference summaries. For instance, Paulus et al. [7] used a hybrid approach combining supervised learning with reinforcement learning to optimize for *ROUGE* scores. Narayan et al. [18] employed RL to jointly optimize for relevance, coherence, and saliency in extractive summarization.

2.4. Hierarchical Reinforcement Learning

Hierarchical Reinforcement Learning (HRL) is an extension of traditional RL that introduces a hierarchy in the decision-making process. In HRL, the learning problem is decomposed into a hierarchy of smaller sub-problems, allowing the agent to learn at multiple levels of temporal abstraction [19]. This approach has shown promise in handling complex, long-horizon tasks in various domains. In the context of NLP, HRL has been applied to tasks such as dialogue generation [20] and document summarization [21]. For summarization, HRL can potentially enable the model to make high-level decisions regarding summary structure and content, while also optimizing low-level decisions regarding word choice and sentence construction.

2.5. Adaptive Summarization

Recent research has recognized the need for adaptive summarization systems that can generate summaries of varying lengths and under different time constraints. Saito et al. [22] introduced a length-controlled summarization model using RL, which can generate summaries of specified lengths while maintaining quality. However, these methods usually focus on adapting to

 doi:10.20944/preprints202503.2300.v1

different length requirements and do not explicitly consider time constraints or the trade-off between summary quality and generation speed. Our objective is to bridge this gap by developing a hierarchical RL framework that can adapt to both length and time constraints while optimizing for summary quality.

2.6. Evaluation of Summarization Systems

The evaluation of the quality of the generated summaries remains a significant challenge in summarization research. Traditional metrics like *ROUGE* [3], which measure n-gram overlap between generated and reference summaries, have been widely used but have known limitations, particularly for abstractive summaries. More recent metrics like *BERTScore* [23] aim to capture semantic similarity beyond lexical overlap.

Human evaluation remains crucial for assessing certain aspects of summary quality such as coherence, relevance, and factual correctness. However, human evaluation is time-consuming and expensive, making it challenging to employ at scale during model development. Our work aims to address these challenges by integrating a comprehensive evaluation framework that combines automated metrics with targeted human evaluation. We also investigate the use of learned reward functions that can potentially capture more nuanced aspects of summary quality.

3. Methodology

Our approach to adaptive text summarization involves combining a hierarchical model architecture with reinforcement learning techniques to create a flexible system that can generate high-quality summaries under varying time constraints. This section describes the key components of our methodology: the hierarchical summarizer model, the reinforcement learning framework, and the training process.

3.1. Hierarchical Summarizer Model

The core of our system is a hierarchical summarizer model based on the T5 architecture [6]. This model consists of two main components:

- T5**-based Encoder-Decoder**: We use a pre-trained T5 model as the base for our summarizer. The T5 model has demonstrated strong performance in various NLP tasks, including summarization, and provides a solid foundation to generate coherent and informative summaries.
- **Level Selector**: We introduce a level selector module that determines the appropriate level of summarization based on the input and time constraints. This module is implemented as a linear layer on top of output from the T5 encoder.

This architecture enables the model to adapt to various summarization requirements by selecting an appropriate level of detail for the summary.

3.2. Reinforcement Learning Framework

We implement three reinforcement learning algorithms to train our hierarchical summarizer. Each of these algorithms is designed to optimize the summarization policy, but they differ in their approach to policy updates and in how they balance exploration and exploitation.

3.2.1. Proximal Policy Optimization (PPO)

PPO uses a clipped objective function to update the policy, which helps prevent excessive policy changes. The PPO loss function is defined as:

$$L(\theta) = \mathrm{E}[\min(r_t(\theta)A_t, \mathrm{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon)A_t)]$$

 doi:10.20944/preprints202503.2300.v1

where $r_t(\theta)$ is the probability ratio between the new and old policies, A_t is the advantage estimate, and ϵ is the clipping parameter.

3.2.2. Advantage Actor-Critic (A2C)

A2C uses an actor-critic architecture, where the actor learns the policy and the critic estimates the value function. The A2C loss function combines the policy loss and value function loss:

$$L(\theta) = L_{\text{policy}}(\theta) + c_1 \cdot L_{\text{value}}(\theta) + c_2 \cdot S[\pi_\theta](s_t)$$

where $L_{\text{policy}}(\theta)$ is the policy loss, $L_{\text{value}}(\theta)$ is the value function loss, $S[\pi_\theta](s_t)$ is an entropy term to encourage exploration, and c_1 and c_2 are coefficients.

3.2.3. Soft Actor-Critic (SAC)

SAC incorporates entropy regularization to encourage exploration and learns a soft Q-function. The SAC objective is to maximize the expected reward while also maximizing entropy:

$$J(\pi) = \mathrm{E}_\pi \sum r_t + \alpha \cdot H(\pi(\cdot \mid s_t))$$

where α is the temperature parameter that determines the importance of entropy.

3.3. Reward Function

The reward function is a critical component of our RL framework. We design a multifaceted reward that considers both the quality of the generated summary and the efficiency of the summarization process:

$$R = \frac{ROUGE + BERTScore}{2} - TimePenalty$$

where:

- *ROUGE* is the average of *ROUGE*-1, *ROUGE*-2, and *ROUGE*-L F1 scores
- *BERTScore* measures semantic similarity between generated summary and reference
- *TimePenalty* is calculated as $\max(0, \frac{\text{generation\_time} - \text{time\_constraint}}{\text{time\_constraint}})$

This reward function balances summary quality, semantic similarity to the reference, and adherence to time constraints.

3.4. Training Process

The training process involves iteratively updating the model based on the rewards received for generated summaries. For each training step:

1. The model generates summaries for a batch of input articles.
2. Rewards are computed based on summary quality and generation time.
3. The RL algorithm updates the model parameters to maximize the expected rewards.

We implement early stopping based on validation performance to prevent overfitting. The training continues until either a maximum number of epochs is reached or the validation performance stops improving for a specified number of consecutive epochs. This approach combines the strengths of hierarchical modeling, reinforcement learning, and adaptive summarization to create a flexible and efficient summarization system. By using RL algorithms to optimize the summarization policy, our system can learn to balance summary quality and generation speed, adapting to different time constraints as necessary.

 doi:10.20944/preprints202503.2300.v1

4. Experimental Setup

To evaluate the effectiveness of our hierarchical reinforcement learning approach to adaptive text summarization, we conducted a series of experiments using the CNN/DailyMail dataset [4], a well-known and widely-used benchmark in text summarization research. This dataset consists of news articles along with human-written summaries, making it ideal for training and evaluating summarization models. For our experiments, we used a subset of the data:

- Training set: 5,000 articles
- Validation set: 500 articles
- Test set: 500 articles

4.1. Implementation Details

We implemented our models using PyTorch and the Hugging Face Transformers library. The T5-small model was used as the base for our hierarchical summarizer. For each of the three reinforcement learning algorithms (PPO, A2C, and SAC), we used the following hyperparameters:

- Learning rate: 3e-5
- Batch size: 4
- Maximum number of epochs: 100
- Early stopping patience: 3 epochs

For the PPO algorithm, we set the clipping parameter ϵ to 0.2. For SAC, we used an initial temperature parameter α of 0.01 and learned this parameter during training.

4.2. Evaluation Metrics

We employed a comprehensive set of evaluation metrics to assess the performance of our models:

- *ROUGE* scores: We computed *ROUGE*-1, *ROUGE*-2, and *ROUGE*-L F1 scores to measure the overlap between generated summaries and reference summaries.
- *BERTScore*: This metric was used to evaluate the semantic similarity between generated and reference summaries.
- Generation Time: We measured the time taken to generate each summary to evaluate the model's efficiency and adherence to time constraints.

4.3. Experimental Procedure

For each RL algorithm (PPO, A2C, and SAC), we trained the model for a maximum of 100 epochs or until early stopping was triggered. During training, we randomly assigned time constraints to each article to simulate varying real-world scenarios. We evaluated the models on the test set under three different time constraint settings: *short* (1 second), *medium* (2 seconds), and *long* (3 seconds). This enabled us to evaluate the models' ability to adjust to varying time constraints while maintaining the quality of the summary.

5. Results and Discussion

5.1. Quantitative Results

We evaluated the performance of our three reinforcement learning models (PPO, A2C, and SAC) on the test set of 500 articles from the CNN/DailyMail dataset. The results are presented in Table 1, showing the average *ROUGE* scores, *BERTScore*, and test reward [1] for each model.

---

[1] Individual *ROUGE* and *BERTScore* values for the SAC model were not reported in the output.

 doi:10.20944/preprints202503.2300.v1

Table 1. Performance of different models on the test set.

Model	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Test Reward
PPO	0.1222	0.0527	0.0789	0.7519	-0.4853
A2C	0.0629	0.0202	0.0336	0.6139	-0.4597
SAC	*	*	*	*	-2.1521

To provide a more comprehensive view of our results, Figures 1, 2, and 3 illustrate the performance of each algorithm across different metrics and summary levels.

Figure 4 provides a direct comparison of *ROUGE*-1 scores, *BERTScore*s, and generation time distributions across all three algorithms for summary level 0. This figure clearly illustrates the superior performance of PPO in terms of *ROUGE*-1 and *BERTScore*, while A2C shows a wider distribution of scores. The SAC algorithm's poor performance is evident in its *ROUGE*-1 and *BERTScore* distributions. Interestingly, all three algorithms show similar generation time distributions, with SAC having a slightly tighter distribution.

Key observations from the quantitative results:

1. The PPO model achieved the highest *ROUGE* scores and *BERTScore* among the three models, indicating that it produced summaries that were most similar to the reference summaries in terms of content overlap and semantic similarity.
2. The A2C model showed lower *ROUGE* scores and *BERTScore* compared to PPO, suggesting that its summaries were less similar to the reference summaries.
3. Interestingly, despite having lower *ROUGE* and *BERTScore* values, the A2C model achieved a slightly better (less negative) test reward than the PPO model. This suggests that the A2C model may have been more effective at balancing summary quality with generation time and adherence to time constraints.
4. The SAC model had the lowest test reward among the three models. The figures reveal that this was due to extremely low *ROUGE* scores and *BERTScore*s across all summary levels.
5. The overall negative test rewards for all models indicate that there is room for improvement in balancing summary quality with generation time constraints.

 doi:10.20944/preprints202503.2300.v1

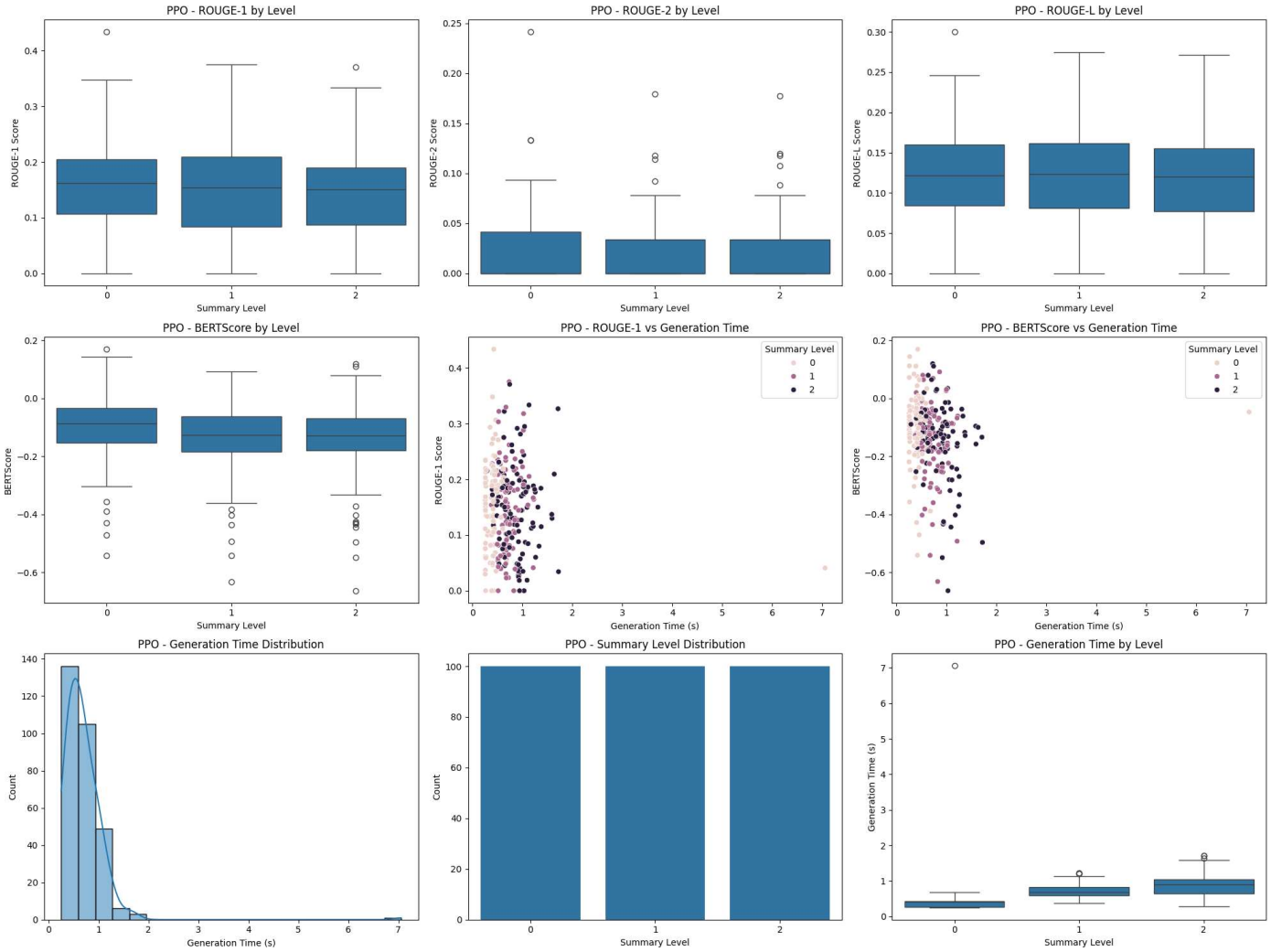

Figure 1. Performance metrics for PPO algorithm across different summary levels. We can observe that *ROUGE* scores and *BERTScores* vary across summary levels, with a slight trend towards higher scores for longer summaries (level 2). The generation time increases with summary level, as expected.

 doi:10.20944/preprints202503.2300.v1

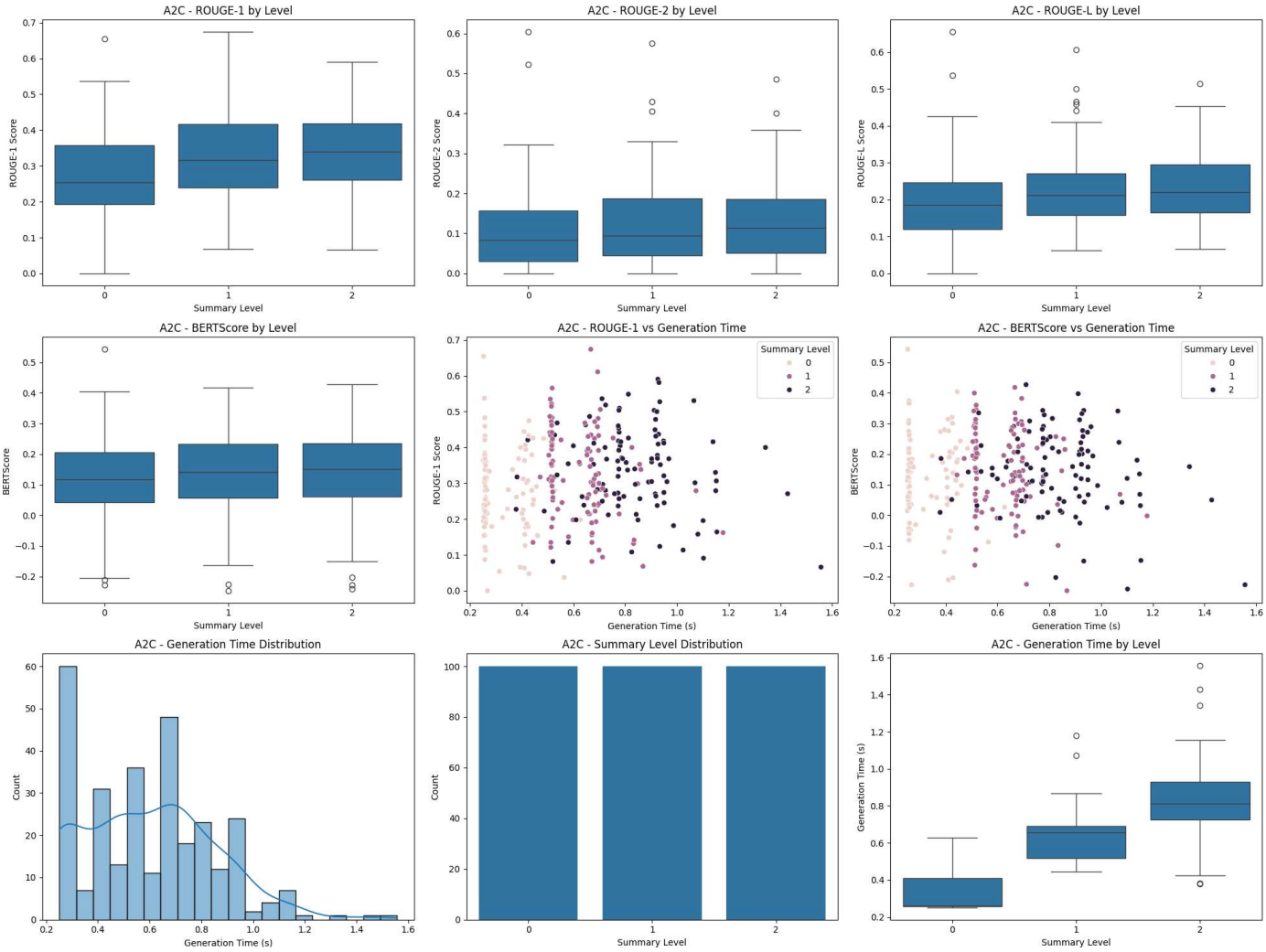

Figure 2. Performance metrics for A2C algorithm across different summary levels. The A2C algorithm shows a more pronounced increase in *ROUGE* scores and *BERTScore*s for higher summary levels compared to PPO. The generation time also increases with summary level, but with a steeper slope than PPO.

 doi:10.20944/preprints202503.2300.v1

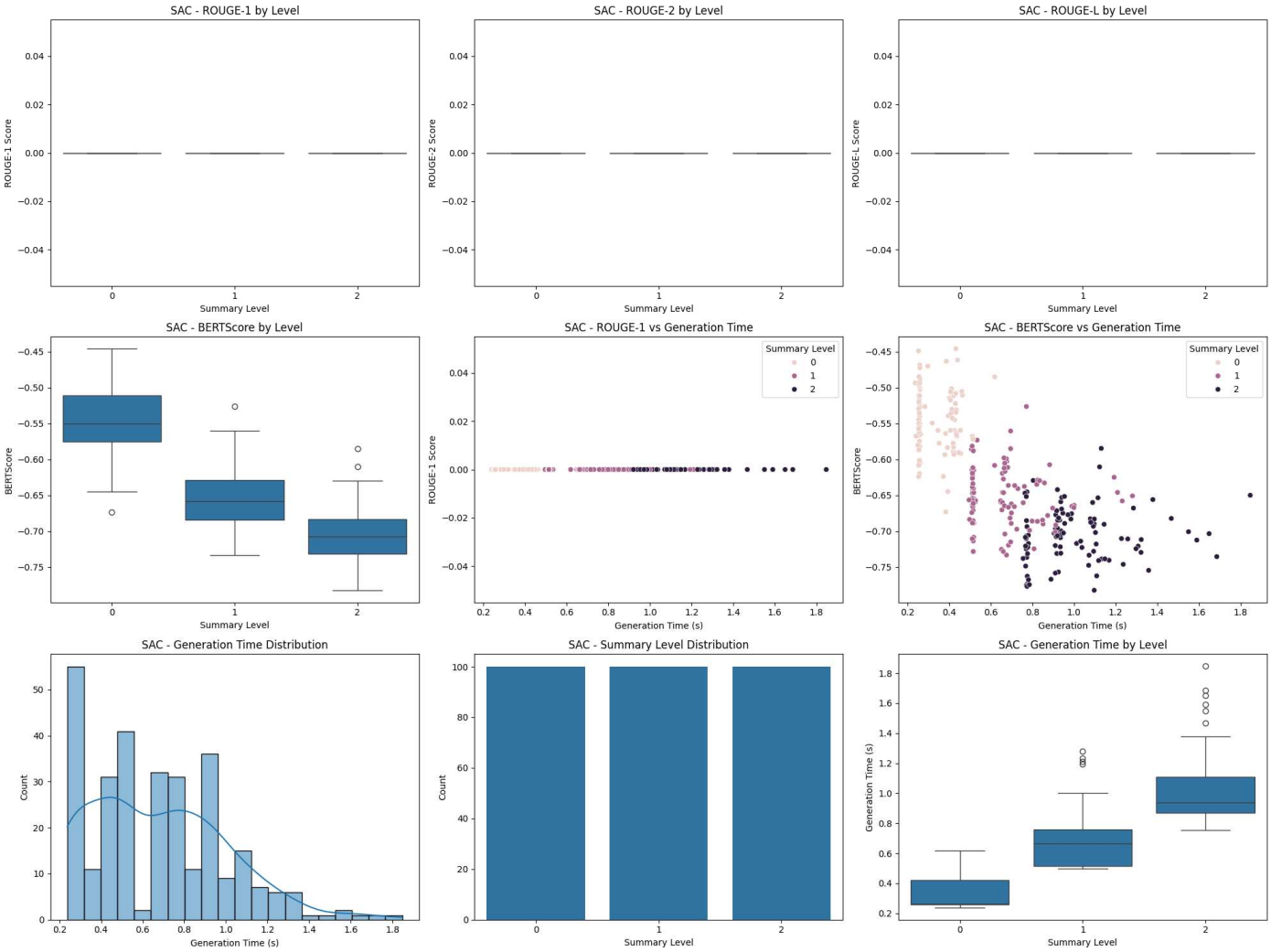

Figure 3. Performance metrics for SAC algorithm across different summary levels. Interestingly, the SAC algorithm shows very low *ROUGE* scores across all summary levels, which aligns with its poor test reward. However, it maintains a similar pattern of increasing generation time with summary level.

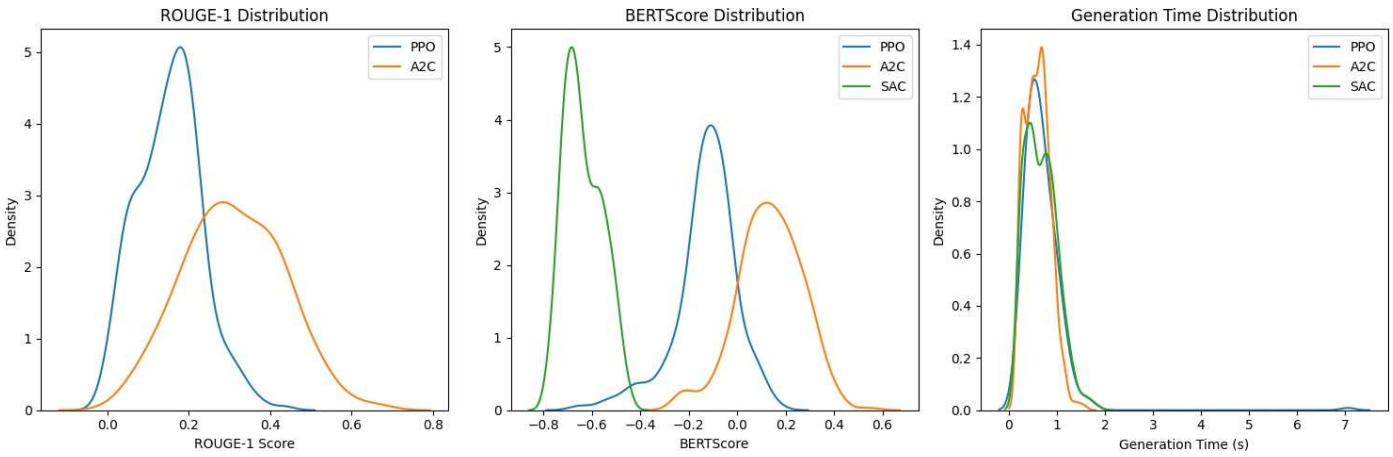

Figure 4. Comparison of *ROUGE*-1, *BERTScore*, and Generation Time distributions across algorithms for summary level 0.

5.2. Discussion

Our experimental results provide insights into the performance of different reinforcement learning algorithms for adaptive text summarization:

 doi:10.20944/preprints202503.2300.v1

1. Trade-off between quality and speed: The discrepancy between *ROUGE* scores and test rewards, particularly for the A2C model, highlights the complex trade-off between summary quality and generation speed. While PPO produced summaries more similar to the references, A2C seems to have found a better balance between quality and time constraints.
2. Challenges with SAC: The significantly lower test reward for the SAC model is clearly explained by its poor performance in *ROUGE* and *BERTScore* metrics, as shown in Figures 3 and 4. This suggests that the SAC algorithm struggled to adapt to the summarization task or to effectively balance the multiple objectives (quality, semantic similarity, and time constraints). Further investigation would be needed to understand the reasons for its underperformance.
3. Adaptation to summary levels: Figures 1, 2, and 3 show how each algorithm adapts to different summary levels. Generally, higher summary levels (longer summaries) tend to achieve better *ROUGE* and *BERTScore*s, but at the cost of longer generation times. This illustrates the challenge of balancing summary quality with time constraints.
4. Room for improvement: The overall low *ROUGE* scores and negative test rewards indicate that there is substantial room for improvement in our models. This could involve refining the reward function, adjusting the model architecture, or exploring alternative reinforcement learning algorithms.
5. Limitations of evaluation metrics: The discrepancy between *ROUGE* scores and test rewards underscores the limitations of relying solely on *ROUGE* for evaluating summarization quality, especially in an adaptive setting where generation time is also a factor. The inclusion of BERTScore provides additional insight, but future work might explore more comprehensive evaluation metrics.

These results demonstrate the potential of using reinforcement learning for adaptive text summarization while also highlighting the challenges involved. The PPO algorithm showed the best performance in terms of traditional summarization metrics, but the A2C algorithm achieved a better balance of quality and efficiency according to our reward function. The poor performance of SAC suggests that not all reinforcement learning algorithms are equally suited to this task, emphasizing the importance of algorithm selection in hierarchical reinforcement learning for text summarization.

6. Conclusion and Future Work

This study introduced a novel approach to adaptive text summarization using hierarchical reinforcement learning. We implemented and compared three reinforcement learning algorithms - Proximal Policy Optimization (PPO), Advantage Actor-Critic (A2C), and Soft Actor-Critic (SAC) - in the context of generating summaries under varying time constraints. Our experiments on the CNN/DailyMail dataset yielded several key findings:

1. Reinforcement learning can be effectively applied to the task of adaptive summarization, allowing models to balance summary quality with generation time.
2. The PPO algorithm demonstrated the best performance in terms of traditional summarization metrics (ROUGE scores and *BERTScore*), suggesting its effectiveness in generating summaries that closely match reference summaries.
3. The A2C algorithm achieved a slightly better overall test reward despite lower *ROUGE* scores, indicating a potentially better balance between summary quality and adherence to time constraints.
4. The SAC algorithm underperformed compared to PPO and A2C, highlighting the challenges of applying this approach to the summarization task.
5. All models showed room for improvement, as evidenced by the overall low *ROUGE* scores and negative test rewards.

These results demonstrate both the potential and the challenges of using reinforcement learning for adaptive text summarization. While our approach shows promise in creating a flexible summarization system that can adapt to different time constraints, there are clear areas for

 doi:10.20944/preprints202503.2300.v1

improvement and further research. Based on our findings, we propose several directions for future work:

6.1. Reward Function Refinement

Given the discrepancies between *ROUGE* scores and overall rewards, future work could focus on developing more sophisticated reward functions that better capture the balance between summary quality and generation efficiency. This could involve incorporating more nuanced metrics of summary quality or exploring learned reward functions.

6.2. Model Architecture Improvements

While we used a `T5`-based model for our summarizer, future work could explore other architectures or ways to more effectively integrate the level selector with the base summarization model.

6.3. Hyperparameter Tuning

A more extensive search of hyperparameters for each RL algorithm could potentially improve performance, especially for the underperforming `SAC` model.

6.4. Exploration of Other RL Algorithms

While we compared three popular RL algorithms, future work could investigate other approaches, such as Deep Deterministic Policy Gradient (DDPG) or multi-agent RL techniques.

6.5. Larger-Scale Experiments

Conducting experiments on a larger subset of the CNN/DailyMail dataset or other summarization datasets could provide more robust insights into the performance and generalizability of our approach.

6.6. Analysis of Generated Summaries

A qualitative analysis of the summaries generated by each model could provide insights into their strengths and weaknesses, guiding future improvements.

6.7. Time Constraint Adaptation

More detailed analysis of how well each model adapts to different time constraints could help in understanding the true flexibility of our approach.

6.8. Human Evaluation

Incorporating human evaluation of summary quality could provide valuable insights beyond what automated metrics can capture, especially in assessing the readability and coherence of generated summaries.

6.9. Application to Other Domains

While we focused on news article summarization, future work could explore the applicability of our approach to other domains, such as scientific literature or legal documents, where adaptive summarization could be particularly valuable.

By addressing these areas, future research can build upon our work to develop more effective and flexible adaptive summarization systems. Such systems have the potential to significantly enhance information access and understanding across various domains, from journalism and research to business intelligence and decision-making processes.

 doi:10.20944/preprints202503.2300.v1

Acknowledgments: The author would like to thank the Laboratory for Analytic Sciences (LAS) at North Carolina State University for providing the resources and support necessary to conduct this research.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Nallapati, R.; Zhou, B.; Gulcehre, C.; Xiang, B.; et al. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023* **2016**.
2. Liu, Y.; Lapata, M. Text summarization with pretrained encoders. *arXiv preprint arXiv:1908.08345* **2019**.
3. Lin, C.Y. Rouge: A package for automatic evaluation of summaries. In Proceedings of the Text summarization branches out, 2004, pp. 74–81.
4. Hermann, K.M.; Kocisky, T.; Grefenstette, E.; Espeholt, L.; Kay, W.; Suleyman, M.; Blunsom, P. Teaching machines to read and comprehend. *Advances in neural information processing systems* **2015**, *28*.
5. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. Bart: Denoising sequence- to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461* **2019**.
6. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research* **2020**, *21*, 1–67.
7. Paulus, R.; Xiong, C.; Socher, R. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304* **2017**.
8. El-Kassas, W.S.; Salama, C.R.; Rafea, A.A.; Mohamed, H.K. Automatic text summarization: A comprehensive survey. *Expert systems with applications* **2021**, *165*, 113679.
9. Luhn, H.P. The automatic creation of literature abstracts. *IBM Journal of research and development* **1958**, *2*, 159–165.
10. Khan, R.; Qian, Y.; Naeem, S. Extractive based text summarization using k-means and tf-idf. *International Journal of Information Engineering and Electronic Business* **2019**, *12*, 33.
11. Nallapati, R.; Zhai, F.; Zhou, B. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In Proceedings of the Proceedings of the AAAI conference on artificial intelligence, 2017, Vol. 31.
12. Erkan, G.; Radev, D.R. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research* **2004**, *22*, 457–479.
13. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. *Advances in neural information processing systems* **2014**, *27*.
14. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Advances in neural information processing systems* **2017**, *30*.
15. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* **2014**.
16. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* **2018**.
17. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Advances in neural information processing systems* **2020**, *33*, 1877–1901.
18. Narayan, S.; Cohen, S.B.; Lapata, M. Ranking sentences for extractive summarization with reinforcement learning. *arXiv preprint arXiv:1802.08636* **2018**.
19. Barto, A.G.; Mahadevan, S. Recent advances in hierarchical reinforcement learning. *Discrete event dynamic systems* **2003**, *13*, 341–379.

 doi:10.20944/preprints202503.2300.v1

20. Saleh, A.; Jaques, N.; Ghandeharioun, A.; Shen, J.; Picard, R. Hierarchical reinforcement learning for open-domain dialog. In Proceedings of the Proceedings of the AAAI conference on artificial intelligence, 2020, Vol. 34, pp. 8741–8748.
21. Wu, Y.; Hu, B. Learning to extract coherent summary via deep reinforcement learning. In Proceedings of the Proceedings of the AAAI conference on artificial intelligence, 2018, Vol. 32.
22. Saito, I.; Nishida, K.; Nishida, K.; Otsuka, A.; Asano, H.; Tomita, J.; Shindo, H.; Matsumoto, Y. Length-controllable abstractive summarization by guiding with summary prototype. *arXiv preprint arXiv:2001.07331* **2020**.
23. Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K.Q.; Artzi, Y. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675* **2019**.