Comparison of Prompt Variations on Poetry Quality Using a Fine-Tuned GPT-2 Model

I Wayan Ivan Zenatmaja[1], Tirana Noor Fatyanosa[1⋆], Prima Zulvarina[1], and Heru Nurwarsito[1]

[1] Faculty of Computer Science, Brawijaya University, Malang, Indonesia
iwayanivanz@student.ub.ac.id, fatyanosa@ub.ac.id, primazulvarina@ub.ac.id, heru@ub.ac.id

Abstract. Natural Language Generation (NLG) enables generative models such as GPT-2 for Indonesian poetry generation, yet challenges remain in fluency and diversity for low-resource languages. This study analyzes the effects of prompt variations (zero-shot, one-shot, and few-shot), hyperparameters, and fine-tuning on generated poetry quality. Three models were evaluated: GPT-2 Small Indonesia (pre-trained), GPT-2 Small Indonesia Fine-Tuning Poem (pre-trained), and GPT-2 FT (further fine-tuned on 125 poems by Goenawan Mohamad). Perplexity and distinct n-gram (distinct-1, distinct-2) were used to measure fluency and lexical diversity. The results reveal a pronounced trade-off: GPT-2 FT achieved the lowest perplexity (1.907) under one-shot prompting, which suggests highest predictive confidence, but also produced the lowest lexical diversity (distinct-1 = 0.1146, distinct-2 = 0.1809), a classic example of degenerative low-entropy output. Conversely, the highest distinct unigram (0.9169) and bigram (0.9809) scores were obtained by pre-trained models. These findings demonstrate that prompt variation, hyperparameter tuning, and fine-tuning significantly influence the fluency-diversity trade-off.

Keywords: GPT-2 · Fine-Tuning · Poetry Generation · Prompt Variation · Indonesian Language

1 Introduction

Poetry functions as both a medium for esthetic expression and a means of conveying the values, emotions, and culture of a society. The uniqueness of its linguistic structure and depth of meaning make poetry challenging for machines to replicate. Advances in Natural Language Processing (NLP) have spurred generative language models such as GPT-2 to mimic human creativity, including poetry generation, although quality and esthetic authenticity remain debated [19].

A growing body of research indicates that while AI-generated poetry can be coherent and sometimes indistinguishable from human-written works, the main

⋆ Corresponding author.

© The Author(s) 2026
M. Murtadha et al. (eds.), *Proceedings of the 2026 11th International Conference on Sustainable Information Engineering and Technology (SIET 2026)*, Atlantis Highlights in Computer Sciences 26,
https://doi.org/10.2991/978-94-6239-828-3_12

issue remains the lack of variation and depth [7]. Poetry generation systems often lack diversity in structure, vocabulary, and semantics, resulting in repetitive output [2]. Efforts to address this include controlling style, rhyme, and meter [18,27], as well as prompt engineering strategies such as zero-shot, one-shot, and few-shot prompting [4].

Unfortunately, most studies focus on high-resource languages such as English or Mandarin, while research on Indonesian remains limited. Indonesian is a low-resource language for NLP due to limited public corpora [12]. Siallagan and Alfina [22] demonstrate that GPT-2 outperforms SeqGAN in Indonesian *pantun* generation, particularly in structure, rhyme, and lexical richness; however, their work still requires comprehensive quantitative evaluation. Gutiérrez-Megías et al. [9] highlight perplexity as an effective metric to distinguish human-written from machine-generated text, while Chakrabarty et al. [6] show that instruction-based approaches improve poetry quality in terms of rhyme, lexical diversity, and naturalness. Zhang and Eger [28] emphasize diversity through a multi-agent social learning framework.

Nevertheless, Sawicki et al. [21] note limitations of prompting alone: GPT models struggle to mimic a poet's distinctive style even with multiple examples, making fine-tuning more effective for stylistic consistency. Thus, this study addresses the gap by comparing the effects of prompt variations on Indonesian poetry using fine-tuned GPT-2, examining the comparative effects of prompts and hyperparameters (temperature, top-k, top-p, repetition penalty) on generation quality, evaluated via perplexity and lexical diversity.

2 Methodology

This study employs a quantitative approach to evaluate poetry generated by GPT-2 Fine-Tuned (GPT-2 FT). The process consists of data collection, model initialization, fine-tuning, and prompt-based generation.

Data Collection. The dataset comprises 125 poems by Goenawan Mohamad, extracted from *Sajak-sajak Goenawan Mohamad dan Sajak-sajak Taufik Ismail* (Poems by Goenawan Mohamad and Poems by Taufik Ismail) [25] (22 poems via Pdfplumber) and *70 Puisi* (70 Poems) [16] (103 poems via EasyOCR with regex segmentation). Data were structured in JSON with fields `title`, `page`, `year`, and `text`, standardized to lowercase with numbers and punctuation removed. The dataset was split 90/10 for training and evaluation (random seed 42). This study is constrained by the limited size and single-author scope.

Model Initialization. Three models based on GPT-2 were evaluated (Table 1). The first serves as a general-domain baseline, the second as an existing poetry-fine-tuned comparative baseline, and the third as our proposed model further fine-tuned on Goenawan Mohamad's poetry using the SFT-Trainer framework.

Table 1. Model Configurations for Poetry Generation

Model Name	Status	Role
GPT-2 Small Indonesia [5]	Pre-trained	Baseline
GPT-2 Small Indonesia Fine-Tuning Poem [1]	Existing poetry-fine-tuned	Comparative baseline
GPT-2 FT [11]	Further fine-tuned from GPT-2 Small Indonesia Fine-Tuning Poem	Proposed model

Fine-Tuning. The GPT-2 Small Indonesia Fine-Tuning Poem was further fine-tuned using SFTTrainer on the collected dataset, yielding GPT-2 FT. Training configuration: block size 512, 5 epochs, batch size 4, 2 gradient accumulation steps, learning rate 3e−5, as shown in Table 2.

Table 2. Training Configuration for GPT-2 FT

Hyperparameter	Value
Block size	512
Epochs	5
Batch size	4
Gradient accumulation steps	2
Learning rate	3×10^{-5}

Prompt-Based Generation. Prompt strategies follow [21], translated into Indonesian. Templates consist of structured input–output pairs: a `PROMPT` containing a brief poem summary, followed by a `COMPLETION` representing the full poem. For each prompt, the model generates 20 poem samples. Table 3 presents the consolidated templates. Prompt strategies are divided into three types: zero-shot, one-shot, and few-shot. Zero-shot provides only a task description; one-shot adds a single example; few-shot provides several examples before generation [15,17]. The few-shot instruction spans 1,341 tokens, whereas the one-shot comprises 352 tokens.

Generation hyperparameters including top-k, top-p, temperature, repetition penalty, and max tokens were configured during initialization to control the diversity and repetition of the generated poem text. The top-k and the top-p (nucleus sampling) suppress the unreliable tail of the probability distribution by limiting choices to the k most likely tokens or to the smallest token set whose cumulative probability reaches threshold p [3,10]. Temperature (T) scales logits before the softmax layer to control output randomness: lower values increase determinism, while higher values encourage lexical creativity [20,14]. Repetition penalty mitigates unnatural phrase repetition by penalizing the logit scores of previously generated tokens [3,14]. Max tokens was set to ensure generation stops

at a poem-relevant point, as increasing output length does not always correlate with quality improvement [26].

Table 3. Prompting Strategies: Types and Templates

Prompt Types	Template
Zero-Shot	*Tulislah puisi dengan gaya {PENYAIR}.* *Puisi tersebut harus sesuai dengan ringkasan berikut:* *{RINGKASAN PUISI}* `Write a poem in the style of {POET}.` `The poem must follow the summary below:` `{POEM SUMMARY}`
One-Shot	*PROMPT:* *{RINGKASAN PUISI 1}* *COMPLETION:* *{ISI PUISI 1}* *PROMPT:* *{RINGKASAN PUISI YANG AKAN DIHASILKAN,* *DARI DATASET GOENAWAN MOHAMAD}* *COMPLETION:* `PROMPT:` `{POEM SUMMARY 1}` `COMPLETION:` `{POEM CONTENT 1}` `PROMPT:` `{POEM SUMMARY TO BE GENERATED,` `FROM GOENAWAN MOHAMAD DATASET}` `COMPLETION:`
Few-Shot	*PROMPT:* *{RINGKASAN PUISI 1}* *COMPLETION:* *{ISI PUISI 1}* *.....* *PROMPT:* *{RINGKASAN PUISI 3}* *COMPLETION:* *{ISI PUISI 3}* *PROMPT:* *{RINGKASAN PUISI YANG AKAN DIHASILKAN,* *DARI DATASET GOENAWAN MOHAMAD}* *COMPLETION:* `PROMPT:` `{POEM SUMMARY 1}` `COMPLETION:` `{POEM CONTENT 1}` `.....` `PROMPT:` `{POEM SUMMARY 3}` `COMPLETION:` `{POEM CONTENT 3}` `PROMPT:` `{POEM SUMMARY TO BE GENERATED,` `FROM GOENAWAN MOHAMAD DATASET}` `COMPLETION:`

3 Hyperparameter Tuning and Results

Table 4 presents hyperparameter tuning results. Zero-shot and one-shot settings consistently yield lower perplexity than few-shot across all architectures. In terms of lexical diversity, one-shot achieves the highest distinct unigram scores

for most architectures, while one-shot and few-shot show relatively high distinct bigram scores.

Table 4. Hyperparameter Tuning Result

Model	Prompt Var.	Hyperparameter					PPL (Mean) ↓	Dist-1 (Mean) ↑	Dist-2 (Mean) ↑
		Max	Temp	k	p	Pen.			
GPT-2 Small Indonesia	Zero-shot	229	0.7	50	0.90	1.10	12.110	0.7506	0.6788
		258	0.9	50	0.95	1.15	26.259	0.8301	0.9614
		1012	0.6	40	0.85	1.20	2.668	0.2315	0.3644
	One-shot	229	0.7	50	0.90	1.10	40.883	0.8191	0.8970
		258	**0.9**	**50**	**0.95**	**1.15**	19.230	0.8695	**0.9809**
		1012	0.6	40	0.85	1.20	2.144	0.2432	0.2741
	Few-shot	1012	0.7	50	0.90	1.10	19.416	0.5714	0.9281
		1012	0.9	50	0.95	1.15	29.089	0.6726	0.9768
		1012	0.6	40	0.85	1.20	16.745	0.5559	0.9102
GPT-2 Small Indo Poem	Zero-shot	229	0.7	50	0.90	1.10	10.440	0.5466	0.6772
		258	0.9	50	0.95	1.15	22.034	0.8119	0.9622
		1012	0.6	40	0.85	1.20	2.684	0.1611	0.2811
	One-shot	**229**	**0.7**	**50**	**0.90**	**1.10**	163.746	**0.9169**	0.9576
		258	0.9	50	0.95	1.15	47.236	0.8668	0.9703
		1012	0.6	40	0.85	1.20	3.004	0.2255	0.3768
	Few-shot	1012	0.7	50	0.90	1.10	27.183	0.5789	0.9334
		1012	0.9	50	0.95	1.15	29.389	0.6343	0.9731
		1012	0.6	40	0.85	1.20	24.340	0.5283	0.8892
GPT-2 FT	Zero-shot	229	0.7	50	0.90	1.10	13.391	0.7001	0.8200
		258	0.9	50	0.95	1.15	23.707	0.7726	0.9189
		1012	0.6	40	0.85	1.20	2.224	0.1668	0.2694
	One-shot	229	0.7	50	0.90	1.10	92.106	0.8572	0.8986
		258	0.9	50	0.95	1.15	33.043	0.8457	0.9482
		1012	**0.6**	**40**	**0.85**	**1.20**	**1.907**	0.1146	0.1809
	Few-shot	1012	0.7	50	0.90	1.10	24.043	0.5964	0.9575
		1012	0.9	50	0.95	1.15	28.177	0.6305	0.9806
		1012	0.6	40	0.85	1.20	21.875	0.5977	0.9362

Table 6 summarizes the best configurations. GPT-2 FT achieved the lowest perplexity (1.907) under one-shot prompt with max tokens 1012, temperature 0.6, top-k 40, top-p 0.85, and repetition penalty 1.2. This is consistent with [9]: lower perplexity indicates better predictive confidence. However, this configuration simultaneously yielded the lowest lexical diversity (distinct-1 = 0.1146, distinct-2 = 0.1809), which produced degenerative low-entropy output. Decoding

Table 5. Sample Poetry Generation Outputs

Model	Instruction	Generated Excerpt
GPT-2 Small Indonesia (One-shot)	Tuliskan sebuah puisi gaya Goenawan Mohamad. Puisi ini harus sesuai dengan ringkasan berikut: Pada puisi pertama gambarkan suasana malam yang hening dan magis melalui metafora musik obo yang dimainkan secara imajiner. Alam, bunyi, dan kesunyian berpadu membentuk pengalaman estetik yang hampir mistis. Waktu dan realitas menjadi kabur, sementara musik berfungsi sebagai mantra yang menangguhkan pagi, hari, dan kepastian. Puisi ini menekankan pengalaman batin, ekstase sesaat, serta keterasingan manusia di tengah lanskap alam dan suara. Teks puisi pertama: Ada sebuah obo dimainkan hantu hutan dan malam jadi hijau sesenyap lukisan Di bawah bulan ekstase dan ayun awan meremang ritme tenung di pebukitan Musik terbaik, Adik, mungkin membius beratus detik suara genta dan gerimis pada pohon-pohon tropis Maka pagi tak jadi pagi hari tak jadi hari hanya angin dan orkes saling menggores Bahkan nada yang jatuh ke biru kolam dipungut balik burung-burung malam JS. Bach, mantera manakah yang akan mengakhiri ini? ketika esok tak ada lagi? Write a poem in the style of Goenawan Mohamad. This poem must follow the summary below: In the first poem, depict a quiet and magical night atmosphere through the metaphor of imaginary oboe music. Nature, sound, and silence blend to form an almost mystical aesthetic experience. Time and reality blur, while music functions as a mantra that suspends morning, day, and certainty. This poem emphasizes inner experience, momentary ecstasy, and human alienation amidst the landscape of nature and sound. First poem text: There is an oboe played by the forest ghost and night turns green as silent as a painting. Under the ecstatic moon and swaying clouds, a trance rhythm trembles across the hills. The best music, Dear, perhaps intoxicates for hundreds of seconds the sound of bells and drizzle on tropical trees. Thus morning does not become morning, day does not become day, only wind and orchestra scratching each other. Even notes that fall into the blue pond are picked back up by night birds. JS. Bach, which mantra will end this? when tomorrow no longer exists?	*Apakah ada jawaban pada pertanyaan di atas? Jawaban pada dua pertanyaan itu hanya satu. Tetapi mengapa sajak-sajak ini hanya membuat kita gelisah? Jika hanya melihat dua-tiga sajak di bagian awal, apa yang akan saya temukan dalam kehidupan ini.* Is there an answer to the question above? The answer to both questions is only one. But why do these verses only make us restless? If only seeing two or three verses at the beginning, what will I find in this life.
GPT-2 Small Indo. Fine-Tuning Poem (One-shot)		*Pagarku telah rapuh* *Karena kau tak mau merawatku* My fence has withered Because you do not want to care for me
GPT-2 FT (One-shot)		*Tak terasa kita telah bersama* *Kita sering tertawa, bercanda, bercerita, tertawa, bercanda, bercanda, bercanda, bercanda* *Aku tau kau sangat menyayangi ku* *Aku tau kau sangat mencintai aku* *Kau selalu berusaha untuk membahagiakanku* *Kau slalu tersenyum padaku* *Tapi kini kau sudah tidak ada* *Aku merasa kesepian tanpa dirimu* *Aku merindukanmu* *Aku rindu kamu* *Aku ingin memilikimu* *Aku ingin memilikimu* *Aku ingin memiliki mu* Unwittingly we have been together. We often laugh, joke, tell stories, laugh, joke, joke, joke, joke. I know you love me very much. I know you love me very much. You always try to make me happy. You always smile at me. But now you are gone. I feel lonely without you. I miss you. I miss you. I want to have you. I want to have you. I want to own you.

with overly peaky distributions (temperature 0.6, top-p 0.85) causes representation degeneration, where the model repeatedly selects tokens from the same cluster [23]. Zhu et al. [30] formalize the self-reinforcement effect: each generated token increases the probability of similar tokens at subsequent steps. At low temperatures, this effect is not mitigated by sampling stochasticity, causing repetitive trajectories. Fine-tuning on a limited single-author dataset (125 poems) renders the representation space anisotropic, which leads the model to memorizing dominant patterns [13]. Thus, the very low PPL shows excessive confidence in repetitive patterns rather than a genuine lexical richness.

Table 6. Best Hyperparameter Tuning Results

Model	Prompt	PPL↓	Dist-1↑	Dist-2↑
GPT-2 Small Indonesia	One-shot	19.230	0.8695	0.9809
GPT-2 Small Indo. Poem	One-shot	163.746	0.9169	0.9576
GPT-2 FT	One-shot	1.907	0.1146	0.1809

For distinct unigram, GPT-2 Small Indonesia Fine-Tuning Poem achieved the highest score (0.9169) under one-shot prompt with max tokens 229, temperature 0.7, top-k 50, top-p 0.9, and repetition penalty 1.1. A repetition penalty of 1.1 effectively promotes unique words by reducing re-selection probability, while temperature 0.7 balances diversity and coherence [8]. For distinct bigram, GPT-2 Small Indonesia achieved the highest score (0.9809) under one-shot prompt with temperature 0.9, top-p 0.95, and repetition penalty 1.15, allowing broader word-pair exploration [29].

Table 5 shows representative outputs. The few-shot dilemma [24] manifests itself concretely: the few-shot prompt (1,341 tokens) substantially exceeds the model's 128 token training sequence length, pushing the sequence into out-of-distribution territory. Consequently, GPT-2 Small strains to maintain coherent self-attention over contexts longer than its training horizon, causing one-shot to consistently outperform few-shot across all metrics.

4 Conclusion

This study examined the effect of hyperparameter settings and prompt variations on Indonesian poetry generation using GPT-2. The results indicate that temperature, top-k, top-p, and repetition penalty significantly influence fluency and lexical diversity. GPT-2 FT achieved the lowest perplexity (1.907) under one-shot prompting, indicating stable generation, but simultaneously produced the lowest lexical diversity, reflecting degenerative low-entropy output. The highest distinct unigram (0.9169) and bigram (0.9809) scores were obtained by pretrained models under one-shot prompts. One-shot prompting consistently outperformed few-shot because the few-shot prompt length exceeded the model's

training sequence length. Therefore, the generation of high-quality poetry depends on careful calibration of hyperparameters and prompt strategies to balance fluency and diversity. Future work should incorporate human evaluation, statistical validation, expanded multi-author datasets, and entropy-aware decoding strategies.

Acknowledgments. This research was funded by the Faculty of Computer Science, Brawijaya University, through the *Hibah Penelitian Tugas Akhir Mahasiswa FILKOM* (Master's Thesis Research Grant for FILKOM Students) scheme. The authors gratefully acknowledge this support.

References

1. ayameRushia: Indonesian gpt-2 finetuned on indonesian poems (2021), https://huggingface.co/ayameRushia/gpt2-small-indonesia-fine-tuning-poem, accessed: November 10, 2023
2. Badura, M., Lampert, M., Dreżewski, R.: System supporting poetry generation using text generation and style transfer methods. Procedia Computer Science **207**, 3310–3319 (2022). https://doi.org/10.1016/j.procs.2022.09.389
3. Basu, S., Ramachandran, G.S., Keskar, N.S., Varshney, L.R.: Mirostat: A neural text decoding algorithm that directly controls perplexity. In: International Conference on Learning Representations (ICLR) (2021), https://openreview.net/pdf?id=W1G1JZEIy5_
4. Boonstra, L.: Prompt Engineering. Google (September 2024), https://www.kaggle.com/whitepaper-prompt-engineering, contributors: Michael Sherman, Yuan Cao, Erick Armbrust, Anant Nawalgaria, Antonio Gulli, Simone Cammel
5. Cahya: Indonesian gpt2 small model (2023), https://huggingface.co/cahya/gpt2-small-indonesian-522M, accessed: May 2, 2026
6. Chakrabarty, T., Padmakumar, V., He, H.: *Help me write a poem*: Instruction tuning as a vehicle for collaborative poetry writing. In: Goldberg, Y., Kozareva, Z., Zhang, Y. (eds.) Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. pp. 6848–6863. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Dec 2022). https://doi.org/10.18653/v1/2022.emnlp-main.460, https://aclanthology.org/2022.emnlp-main.460/
7. Chen, Y., Gröner, H., Zarrieß, S., Eger, S.: Evaluating diversity in automatic poetry generation. In: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. pp. 19671–19692. Association for Computational Linguistics (2024)
8. Dubois, M., Yvon, F., Piantanida, P.: How sampling affects the detectability of machine-written texts: A comprehensive study. In: Christodoulopoulos, C., Chakraborty, T., Rose, C., Peng, V. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2025. pp. 11369–11387. Association for Computational Linguistics, Suzhou, China (Nov 2025). https://doi.org/10.18653/v1/2025.findings-emnlp.609
9. Gutiérrez Megías, A.J., Ureña-López, L.A., Martínez Cámara, E.: The influence of the perplexity score in the detection of machine-generated texts. In: Mitkov, R., Ezzini, S., Ranasinghe, T., Ezeani, I., Khallaf, N., Acarturk, C., Bradbury, M., El-Haj, M., Rayson, P. (eds.) Proceedings of the First International Conference

on Natural Language Processing and Artificial Intelligence for Cyber Security. pp. 80–85. International Conference on Natural Language Processing and Artificial Intelligence for Cyber Security, Lancaster, UK (Jul 2024), https://aclanthology.org/2024.nlpaics-1.10/
10. Holtzman, A., Buys, J., Du, L., Forbes, M., Choi, Y.: The curious case of neural text degeneration. In: International Conference on Learning Representations (ICLR) (2020), https://openreview.net/pdf?id=rygGQyrFvH
11. ivanz3natmaja: Poetry-generation-gpt-2, https://github.com/NLP-UB/Poetry-Generation-GPT-2, accessed 2026-08-03
12. Khairunnisa, S., Chen, Z., Komachi, M.: Dataset enhancement and multilingual transfer for named entity recognition in the indonesian language. ACM Transactions on Asian and Low-Resource Language Information Processing **22**(6), 1–21 (2023). https://doi.org/10.1145/3592854
13. Li, J., Liang, Y., Li, S., Liu, W., Shao, J.: Evade the trap of mediocrity: Promoting diversity and novelty in text generation via concentrating attention. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics (2022)
14. Li, L., Sleem, L., Gentile, N., Nichil, G., State, R.: Exploring the impact of temperature on large language models: Hot or cold? Procedia Computer Science **264**, 242–251 (2025)
15. Li, Y., Bonatti, R., Abdali, S., Wagle, J., Koishida, K.: Data generation using large language models for text classification: An empirical case study (2024), https://arxiv.org/abs/2407.12813
16. Mohamad, G.: 70 Puisi. GRAFITI (2011)
17. Nadăş, M., Dioşan, L., Tomescu, A.: Synthetic data generation using large language models: Advances in text and code. IEEE Access **13**, 134615–134633 (2025). https://doi.org/10.1109/ACCESS.2025.3589503
18. Ormazabal, A., Artetxe, M., Agirrezabal, M., Soroa, A., Agirre, E.: Poelm: A meter- and rhyme-controllable language model for unsupervised poetry generation. In: Findings of the Association for Computational Linguistics: EMNLP 2022. pp. 3652–3667. Association for Computational Linguistics (2022). https://doi.org/10.18653/v1/2022.findings-emnlp.268
19. Porter, B., Machery, E.: Ai-generated poetry is indistinguishable from human-written poetry and is rated more favorably. Scientific Reports **14**(1), 26133 (2024). https://doi.org/10.1038/s41598-024-76900-1
20. Renze, M., Guven, E.: The effect of sampling temperature on problem solving in large language models. In: Findings of the Association for Computational Linguistics: EMNLP 2024. pp. 7346–7356 (2024), https://arxiv.org/abs/2402.05201
21. Sawicki, P., Grzes, M., Goes, F., Brown, D., Peeperkorn, M., Khatun, A.: Bits of grass: Does gpt already know how to write like whitman? (2023), https://arxiv.org/abs/2305.11064
22. Siallagan, E.A., Alfina, I.: Poetry generation for indonesian pantun: Comparison between seqgan and gpt-2. Jurnal Ilmu Komputer dan Informasi **16**(1), 59–67 (2023). https://doi.org/10.21609/jiki.v16i1.1113
23. Su, Y., Lan, T., Wang, Y., Yogatama, D., Kong, L., Collier, N.: A contrastive framework for neural text generation. In: Advances in Neural Information Processing Systems (NeurIPS) (2022), https://arxiv.org/abs/2202.06417
24. Tang, Y., Tuncel, D., Koerner, C., Runkler, T.: The few-shot dilemma: Over-prompting large language models. In: Proc. 3rd Int. Conf. Foundation and Large Language Models (FLLM). pp. 134–141 (2025)

25. Toda, D.N., Nasution, P.: Sajak-sajak Goenawan Mohamad dan Sajak-sajak Taufik Ismail. Pusat Pembinaan dan Pengembangan Bahasa, Departemen Pendidikan dan Kebudayaan, Jakarta (1984), https://repositori.kemendikdasmen.go.id/23730/1/SAJAK-SAJAK%20GOENAWAN%20MOHAMAD%20DAN%20SAJAK-SAJAK%20TAUFIK%20ISMAIL.pdf
26. Wang, C., Liu, X., Awadallah, A.H.: Cost-effective hyperparameter optimization for large language model generation inference. In: AutoML Conference 2023 (2023), https://openreview.net/forum?id=DoGmh8A39O
27. Wöckener, J., Haider, T., Miller, T., Nguyen, T., Nguyen, T., Pham, M., Belouadi, J., Eger, S.: End-to-end style-conditioned poetry generation: What does it take to learn from examples alone? In: Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature. pp. 57–66. Association for Computational Linguistics, Punta Cana, Dominican Republic (online) (2021)
28. Zhang, R., Eger, S.: Llm-based multi-agent poetry generation in non-cooperative environments. Journal of Language Modelling **13**(2) (2026). https://doi.org/10.15398/jlm.v13i2.432
29. Zhou, Y., Keuper, M., Fritz, M.: Balancing diversity and risk in LLM sampling: How to select your method and parameter for open-ended text generation. In: Che, W., Nabende, J., Shutova, E., Pilehvar, M.T. (eds.) Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 26352–26365. Association for Computational Linguistics, Vienna, Austria (Jul 2025). https://doi.org/10.18653/v1/2025.acl-long.1278, https://aclanthology.org/2025.acl-long.1278/
30. Zhu, L., Wang, Z., Wei, Y., Chen, D., Zhang, H., Chen, Y., Wang, Y., Wei, F.: Penalty decoding: Well suppress the self-reinforcement effect in open-ended text generation. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics (2023)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

