SUMMARIZATION FOR POST-PROCESSING TEXT CONTROL

Wen-Hung Wang
Department of Statistics
Purdue University
West Lafayette, IN 47907, USA
wang6058@purdue.edu

Sz-Wei Charng
Department of Statistics
Purdue University
West Lafayette, IN 47907, USA
scharng@purdue.edu

1 INTRODUCTION

Transformer-based Pre-trained Large Language Models (TP-LLMs) have significantly advanced Natural Language Generation (NLG), leading to rapid progress in Controllable Text Generation (CTG), a prominent subfield within NLG (Zhang et al., 2023). Unlike traditional autoregressive methods, CTG generates text subject to specific constraints by modeling the conditional probability $p(x \mid C)$, where C denotes a set of constraints and x represents the generated sentence (Zhang et al., 2023).

Among various CTG methods, post-processing techniques leveraging TP-LLMs for initial text generation followed by constraint enforcement have gained popularity due to their computational efficiency (Zhang et al., 2023). A notable Bayesian-inspired approach employs an accept-reject mechanism on TP-LLM outputs, ensuring that the conditional probability $p(x \mid C)$ remains invariant (Mireshghallah et al., 2022; Forristal et al., 2023; Qin et al., 2022; Miao et al., 2019).

Previous research by Miao et al. (2019) defined this conditional probability as:

$$p(x \mid C) \propto p(x) \cdot S_C^0(x) \cdots S_C^{|C|}(x), \quad x \in \mathcal{X}$$

where $p(x)$ denotes the intrinsic probability of sentence x, $\mathcal{X}$ represents the set of all sentences, and $S_C^i(x)$ quantifies the degree to which sentence x satisfies the i^{th} constraint. Energy-based language models (LMs) further refined this into a Boltzmann distribution:

$$p(x \mid C) \propto e^{-E_C(x)}, \tag{1}$$

with the energy function defined as $E_C(x) = \sum_{i=1}^{|C|} \alpha_i E_i(x)$, a weighted combination of constraint-related energies, subject to $\sum_{i=1}^{|C|} \alpha_i = 1$ (Mireshghallah et al., 2022; Qin et al., 2022; Forristal et al., 2023). Specifically, Forristal et al. (2023) integrated a prompted TP-LLM, Flan-T5-XXL (Chung et al., 2022), within a Metropolis-Hastings (MH) sampling framework (Metropolis et al., 1953), running multiple chains but ultimately selecting only one optimal sample for inference—a method referred to as BLOCK-MH.

To better leverage the collective information from multiple Markov chains, we propose aggregating accepted samples using the Bidirectional and Auto-Regressive Transformers (BART) model (Lewis et al., 2019). In this study, we specifically investigate whether employing a BART-based summarizer can mitigate the underutilization of information from multiple generated samples. Although our results demonstrate that this summarization approach consistently underperforms BLOCK-MH, it achieves more stable and reliable performance. We conclude by hypothesizing the observed inferior performance of the summarization-based approach and discussing future directions.

The rest of the article is organized as follows. In section 2, we describe the dataset we use for style transfer. Section 3 illustrates the summarization-based approach we are examining. We list and summarize related work. Finally, we present the result of our experiment. The implementation of our methods is publicly available at https://github.com/JackWang0102/CS587-Deep-Learning-Final-Project.

2 Dataset

Since our work builds upon improvements by Forristal et al. (2023) and focuses on the style transfer from modern English to Shakespearean style Engliish, we utilize the same dataset: Shakespeare author imitation dataset (Xu et al., 2012) to ensure direct comparability. The GitHub link to the dataset is `https://github.com/cocoxu/Shakespeare`. The dataset contains 27,797 aligned sentence pairs, each consisting of a modern English sentence and its corresponding original Shakespearean version. We shuffled the data and split it into 80% training, 10% validation, and 10% test sets. For classification tasks, we labeled modern sentences as 0 and original sentences as 1, and saved all datasets as CSV files.

3 Proposed Approach

In this section, we first provide an overview of the original BLOCK-MH algorithm introduced by Forristal et al. (2023), detailed in Algorithm 1. Briefly, BLOCK-MH leverages a prompted TP-LLM as a proposal mechanism to generate candidate rewrites, subsequently employing a Metropolis-Hastings (MH) accept-reject framework to regulate generated texts. Conceptually, repeated iterations of Algorithm 1 effectively approximate sampling from a Boltzmann distribution defined in equation 1. The primary objective is to sample from regions where the energy is minimized, ideally the mode(s) of the distribution, i.e., sampling $x^* = \arg\min_{x\in\mathcal{X}} E_C(x)$. Importantly, BLOCK-MH operates as a population-based Markov Chain Monte Carlo (MCMC) method, executing multiple chains concurrently. However, summarizing results from these parallel chains through a single sampled sentence can result in unstable inference characterized by significant variability in performance metrics.

To address this issue, we introduce Algorithm 2, which diverges from BLOCK-MH by incorporating a distinct *summarization step*. Our approach utilizes BART to synthesize multiple candidate sentences from parallel chains, analogous to the statistical concept of sufficiency. The detailed steps of Algorithm 2 are outlined below.

The rationale behind selecting BART as our summarization model stems from its demonstrated superiority in abstractive summarization tasks (Lewis et al., 2019). Unlike extractive summarizers that merely select existing content, BART generates summaries by integrating information and synthesizing novel content from the provided input sentences. Thus, Algorithm 2 leverages BART's capability for information fusion, in contrast to BLOCK-MH, which inherently employs an extractive summarization strategy by selecting a single sentence and disregarding other potentially valuable information. Detailed experiment setting and results are presented in Section 5.

4 Related Work

Energy-based language models (LMs) for post-processing tasks are particularly relevant to our research, primarily due to their computational efficiency and flexibility in accommodating constraints (Forristal et al., 2023; Mireshghallah et al., 2022; Qin et al., 2022). Mix and Match (M&M), proposed by Mireshghallah et al. (2022), leverages BERT as a proposal distribution for token-level replacements within a Metropolis-Hastings (MH) sampling framework. This approach mirrors Gibbs sampling methods (Geman & Geman, 1984) and aligns closely with the approach by Miao et al. (2019), who utilized token-level operations as well but accommodated variable-length outputs by integrating insertion and deletion.

Building upon this, subsequent studies by Forristal et al. (2023) identified critical limitations in M&M, particularly its dependence on BERT and constraints regarding fixed-length outputs. To overcome these shortcomings, Forristal et al. introduced block-MH sampling, enabling comprehensive sentence rewrites through prompted Flan-T5-XXL (Chung et al., 2022), thus facilitating the generation of variable-length sentences.

Further exploration of gradient-based MCMC methods includes the work by Qin et al. (2022), who employed Langevin Dynamics as the proposal mechanism. They utilized continuous relaxation techniques for textual representations, differentiable constraints, and guided discretization methods to effectively address the inherent non-differentiability of text.

Algorithm 1 BLOCK-MH (Forristal et al., 2023)

```
Input: Number of chains $M$, seed text $x_0$, target density $\pi(x) \propto e^{-E_C(x)}$, proposal density $q(x'|x, w)$ (a prompted LLM), number of iterations $N$, user-specified prompt $w$, burn-in period $B$ (with $B < N$).
Output: The sample $x^*$ with the lowest energy among all samples from every chain.
Initialize each chain with $x_0$.
for $m = 1$ to $M$ do
    $x^{(m)} \leftarrow x_0$
end for
for $t = 1$ to $N$ do
    // Process each chain in parallel:
    for each chain $m = 1, \ldots, M$ in parallel do
        Propose $x' \sim q(x'|x^{(m)}, w)$.
        Compute the acceptance ratio:
            $\alpha = \min\Big\{1, \frac{e^{-E_C(x')}\, q(x^{(m)}|x', w)}{e^{-E_C(x^{(m)})}\, q(x'|x^{(m)}, w)}\Big\}$
        Draw $u \sim \text{Uniform}(0, 1)$.
        if $u \leq \alpha$ then
            $x^{(m)} \leftarrow x'$.
        end if
        Save $x^{(m)}$ at iteration $t$.
    end for
end for
Discard the first $B$ iterations as burn-in and collect the remaining samples:
    $S^{(m)} = \{x_t^{(m)}\}_{t=B+1}^{N}$.
Let $S = \bigcup_{m=1}^{M} S^{(m)}$.
return $x^* = \arg\min_{x \in S} E_C(x)$, $\mathcal{S} \subset \mathcal{X}$.
```

Algorithm 2 Summarization MH

```
Input: Number of chains $M$, seed text $x_0$, target density $\pi(x) \propto e^{-E_C(x)}$, proposal density $q(x'|x, w)$ (a prompted LLM), number of iterations $N$, user-specified prompt $w$, burn-in period $B$ (with $B < N$).
Output: The sample $x^*$ with the lowest energy among all samples from every chain.
Initialize each chain with $x_0$.
for $m = 1$ to $M$ do
    $x^{(m)} \leftarrow x_0$
end for
for $t = 1$ to $N$ do
    // Process each chain in parallel:
    for each chain $m = 1, \dots, M$ in parallel do
        Propose $x' \sim q(x'|x^{(m)}, w)$.
        Compute the acceptance ratio:
            $$\alpha = \min\left\{1, \frac{e^{-E_C(x')}\, q(x^{(m)}|x', w)}{e^{-E_C(x^{(m)})}\, q(x'|x^{(m)}, w)}\right\}$$
        Draw $u \sim \text{Uniform}(0, 1)$.
        if $u \leq \alpha$ then
            $x^{(m)} \leftarrow x'$.
        end if
        Save $x^{(m)}$ at iteration $t$.
    end for
end for
Discard the first $B$ iterations as burn-in and collect the remaining samples:
    $$S^{(m)} = \{x_t^{(m)}\}_{t=B+1}^{N}.$$
Let $S = \bigcup_{m=1}^{M} S^{(m)}$.
return $x^* = \text{BART}(S)$.
```

Additionally, alternative controllable text generation (CTG) methodologies such as fine-tuning and retraining have been thoroughly reviewed by Zhang et al. (2023). Notably, post-processing techniques distinguish themselves by decoupling model training from constraint handling, significantly lowering computational demands, especially pertinent given the increasing scale of modern language models.

BART (Lewis et al., 2019) is a denoising autoencoder built upon the Transformer architecture (Vaswani et al., 2017). It is pretrained using self-supervised learning on arbitrarily corrupted documents, employing corruption strategies such as token masking, deletion, and sentence permutation. Structurally, BART integrates a bidirectional encoder, akin to BERT (Devlin et al., 2019), capturing contextual relationships, with an autoregressive decoder, analogous to GPT (Radford et al., 2019), enabling left-to-right text generation. Its training objective focuses on reconstruction, measuring the model's effectiveness in recovering original documents from their corrupted versions. Lewis et al. (2019) demonstrated that BART excels in summarization tasks, particularly in abstractive summarization, where synthesizing new information rather than mere extraction is crucial, making it a robust model candidate for such tasks.

5 Results

5.1 Experiment setup

In our experiment, the prompt we use is: *Rewrite this sentence in the style of William Shakespeare* and the proposal distribution we use is Flan-T5 base, not Flan-T5 XXL as in Forristal et al. (2023). In Forristal et al. (2023), since $q(x^{(m)}|x^{(m)}, w)$ is high, the authors broke the detailed balance and instead proposed using $q(x^{(m)}|x^{(m)}, w)$ in the numerator of α to facilitate the acceptance of different editing $x' \neq x^{(m)}$. We follow the same setting to break the detailed balance.

We focus on the Shakespeare style transfer task on a sentence, i.e., given a modern English sentence x_0, we wish to rewrite it as a Shakespearean style sentence x^* that is semantically close to x_0 and is as Shakespearean as possible. To implement Algorithm 1 and 2 to achieve this goal, we need to formally define the energy function $E_C(x)$ as follows (Forristal et al., 2023):

$$E_C(x, x_0) = \alpha_1 E_{\text{disc}}(x) + \alpha_2 E_{\text{BERTScore}}(x, x_0), \tag{2}$$

where both α_1 and α_1 are set to 0.5 in our experiment. The Shakespeare style transfer task C has two goals, meaning $|C| = 2$. Therefore, Forristal et al. (2023) employ two expert factors E_{disc} and $E_{\text{BERTScore}}$. The former is a style discriminator E_{disc} evaluating how a given sentence x is Shakespearean, i.e., $-\log p(\text{Shakespearean} \mid x)$. E_{disc} is evaluated by a fine-tuned RoBERTa model (Liu et al., 2019) on the Shakespearean data set whereas the latter measures semantic similarity using the negative rescaled BERT score. Due to the limited computation resource, we use RoBERTa-base from Hugging Face instead.

For evaluation, we follow Forristal et al. (2023) and Krishna et al. (2020) to use $J-$score to evaluate the proposed method and BLOCK-MH. $J-$score measures accuracy (ACC), similarity (SIM), and fluency (FL). $J-$score is defined as

$$J(\text{ACC, SIM, FL}) = \sum_{x \in \mathcal{X}} \frac{\text{ACC}(x) \cdot \text{SIM}(x) \cdot \text{FL}(x)}{|\mathcal{X}|} \tag{3}$$

where $\mathcal{X}$ is the test corpus. ACC is discriminator using the fine-tuned RoBERTa-base aforementioned. SIM is the rescaled BERT score. FL is a RoBERTa-base formality classifier. Both ACC and FL are binary. To reduce computation cost, we randomly choose 5 sentences from the test corpus rather than using the whole test corpus. The experiment was done 10 times where the number of chains is set to 5. We admit that our experiment is very small-scale compared to the original paper. However, we are still able to compare two methods (summarization and BLOCK-MH) and provide insights into these two approaches.

We evaluate two Metropolis–Hastings schemes for Shakespearean style transfer on individual sentences:

1. **BLOCK-MH** as in Forristal et al. Forristal et al. (2023), which breaks detailed balance by replacing the usual denominator term with the proposal probability of the current state.

2. **Summarization-MH**, our variant in which we run multiple short chains and then "summarize" the accepted samples into a single output.

All experiments use the prompt:

> *Rewrite this sentence in the style of William Shakespeare.*

Unlike Forristal et al. (2023), who use Flan-T5 XXL as the proposal distribution, we employ Flan-T5 base for both methods, and we fine-tune a RoBERTa-base discriminator (instead of RoBERTa-XXL).

We define the joint energy

$$E(x; x_0) \;=\; \tfrac{1}{2}\, E_{\text{disc}}(x) \;+\; \tfrac{1}{2}\, E_{\text{BERTScore}}(x, x_0),$$

where

- $E_{\text{disc}}(x) = -\log p(\text{Shakespearean} \mid x)$ is the style-discriminator score from our fine-tuned RoBERTa-base model,
- $E_{\text{BERTScore}}(x, x_0)$ measures semantic fidelity via a negative, rescaled BERTScore.

For each input sentence x_0, we run five parallel chains for ten Metropolis steps each, repeating the experiment with ten random seeds. To reduce compute, we randomly sample five sentences from the test set rather than using the full corpus.

Evaluation Metric. We report the J-score from Krishna et al. Krishna et al. (2020), defined as the average product of accuracy (ACC), semantic similarity (SIM), and fluency (FL) over the test sentences:

$$J \;=\; \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} \bigl[\text{ACC}(x) \times \text{SIM}(x) \times \text{FL}(x)\bigr].$$

Here ACC is the RoBERTa-base discriminator, SIM is the rescaled BERTScore, and FL is a binary fluency label from a RoBERTa-base formality classifier.

5.2 Experimental Results

Figure 1 plots J-scores across ten trials for both methods. BLOCK-MH consistently outperforms Summarization-MH, achieving higher average quality. Summarization-MH yields more stable—but lower—scores, reflecting its "averaging" of multiple samples.

We hypothesize that the inferior performance of Summarization-MH stems from **proposal-generator quality**. Using a smaller Flan-T5 base model (and a lightweight discriminator RoBERTa-base) sometimes produces semantically incoherent edits. When an unacceptable sample is accepted, the summarizer can be "contaminated" by these outliers. In contrast, BLOCK-MH simply selects the single lowest-energy sentence and is less vulnerable to noisy proposals. Here is an example:

> *Example.* With seed text "It is born," some accepted proposals under Summarization-MH were:
>
> > "They returned to this place. They were glad."
>
> which bears little semantic or stylistic resemblance to the original. These aberrant samples degrade the final summary.

5.3 Future Directions

To overcome the limitations identified above, we suggest the following avenues:

- **Richer Proposal Distributions.** Combine multiple LLMs (e.g. a mixture of Flan-T5 variants or other transformer models) to improve proposal diversity and ergodicity.
- **Stronger Proposal Generators.** Leverage larger pre-trained or fine-tuned models (e.g. Flan-T5 XXL or GPT-class LLMs) to reduce the rate of incoherent proposals.
- **Interpretable Summarizers.** Replace the black-box summarization step with a more transparent aggregation mechanism (e.g. weighted sampling or deterministic selection rules).

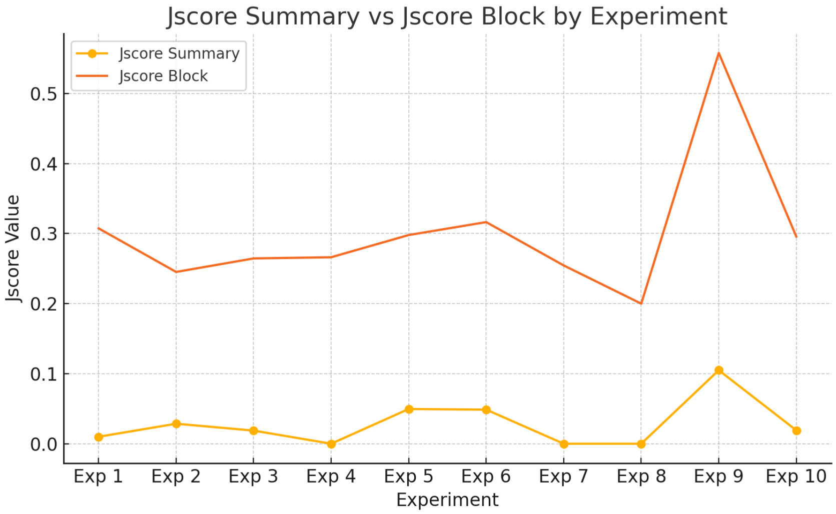

Figure 1: J-scores for BLOCK-MH vs. Summarization-MH over ten trials.

References

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022. URL `https://arxiv.org/abs/2210.11416`.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pp. 4171–4186, 2019.

Jarad Forristal, Niloofar Mireshghallah, Greg Durrett, and Taylor Berg-Kirkpatrick. A block metropolis-hastings sampler for controllable energy-based text generation. *arXiv preprint arXiv:2312.04510*, 2023.

Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6): 721–741, 1984.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. Reformulating unsupervised style transfer as paraphrase generation. *arXiv preprint arXiv:2010.05700*, 2020.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

Ning Miao, Hao Zhou, Lili Mou, Rui Yan, and Lei Li. Cgmh: Constrained sentence generation by metropolis-hastings sampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 6834–6842, 2019.

Fatemehsadat Mireshghallah, Kartik Goyal, and Taylor Berg-Kirkpatrick. Mix and match: Learning-free controllable text generation using energy language models. *arXiv preprint arXiv:2203.13299*, 2022.

Lianhui Qin, Sean Welleck, Daniel Khashabi, and Yejin Choi. Cold decoding: Energy-based constrained text generation with langevin dynamics. *Advances in Neural Information Processing Systems*, 35:9538–9551, 2022.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are few-shot learners. *OpenAI Technical Report*, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Wei Xu, Alan Ritter, Bill Dolan, Ralph Grishman, and Colin Cherry. Paraphrasing for style. In Martin Kay and Christian Boitet (eds.), *Proceedings of COLING 2012*, pp. 2899–2914, Mumbai, India, December 2012. The COLING 2012 Organizing Committee. URL `https://aclanthology.org/C12-1177/`.

Hanqing Zhang, Haolin Song, Shaoyu Li, Ming Zhou, and Dawei Song. A survey of controllable text generation using transformer-based pre-trained language models. *ACM Computing Surveys*, 56(3):1–37, 2023.