Customized Style Transfer using Discrete Sampling

Anugunj Naman
Purdue University
West Lafayette, IN, USA
anaman@purdue.edu

Abstract

Customizing text style or content typically involves extensive fine-tuning of large models, demanding significant data and training. Traditional unsupervised approaches using sampling often yield low diversity and creativity. We present a novel discrete Langevin proposal that samples directly from the categorical token distribution, overcoming these limitations. By adapting the continuous Langevin algorithm for discrete spaces, our approach enables efficient gradient-based sampling. Evaluations on style transfer tasks demonstrate superior performance over state-of-the-art methods in accuracy, BLEU, BERTScore, and diversity. Our proposed approach paves way for advanced customized text generation with desired styles as well as allows future scope for prompt generation for model safeguarding and jail-breaking.

1 Introduction

Customizing text style is an important task in natural language processing that involves generating text conditioned on specific styles or topics (Xu et al., 2012; Gehman et al., 2020; Baheti et al., 2021; Mireshghallah and Berg-Kirkpatrick, 2021). Traditional techniques for tailoring large language models to specific applications typically necessitate extensive fine-tuning on specialized datasets, a process that can be both resource-intensive and inflexible (Keskar et al., 2019; Mai et al., 2020; Gururangan et al., 2020; Chronopoulou et al., 2022). Other approaches avoid extensive retraining by guiding pre-trained models during decoding, blending model-generated likelihoods with heuristic scoring functions (Dathathri et al., 2019; Krause et al., 2021; Yang and Klein, 2021; Goyal et al., 2022). These approaches, however, often require significant modifications to the model architecture or the addition of complex auxiliary modules.

To address these challenges, recent research has focused on improving existing generative strategies. Traditional approaches like Markov chain Monte Carlo (MCMC), including Gibbs sampling, often make minor, localized adjustments to text, which can limit diversity and innovation (Mireshghallah et al., 2022; Kumar et al., 2022). More recently, techniques such as gradient-based Langevin dynamics sampling have been explored to enhance efficiency in continuous spaces (Qin et al., 2022; Kumar et al., 2022). However, these approaches face difficulties such as prompt deviation and mismatches between continuous and discrete representations (Khashabi et al., 2022).

In response to these issues, we propose a novel discrete Langevin dynamics-based approach that facilitates direct sampling from the categorical distribution of tokens inspired by (Zhang et al., 2022) recent work. Our approach enables efficient exploration of the distribution and simultaneous updates of multiple tokens, overcoming the constraints of traditional discretization techniques. We demonstrate that this approach achieves faster convergence and greater output diversity compared to conventional Gibbs and Langevin sampling.

In a series of empirical evaluations, our approach surpasses established techniques like Mix-Match (Mireshghallah and Berg-Kirkpatrick, 2021) and MUCOLA (Kumar et al., 2022) in style transfer and text generation tasks. Our contributions are threefold:

1. Our discrete Langevin approach offers an efficient gradient-based sampler for discrete spaces, achieving robust conditional generation capabilities without requiring additional training. This method outperforms previous Langevin approaches that are limited to continuous spaces.

2. By adjusting multiple tokens simultaneously, it rapidly explores the complex discrete distribution of text compared to single token changes per step, producing diverse outputs.

Proceedings of the 1st Workshop on Customizable NLP: Progress and Challenges in Customizing NLP for a Domain, Application, Group, or Individual (CustomNLP4U), pages 150–155

3. The approach provides a general-purpose sampler that is amenable to customizing text generation across diverse tasks.

2 Related Work

Recent works closely related to our approach include MixMatch and MUCOLA. MixMatch operates within the Energy-Based Model (EBM) framework and employs Gibbs sampling to generate text (Mireshghallah et al., 2022). While this method is effective, it relies on traditional MCMC techniques, which can be slower and less efficient, particularly when applied to discrete data spaces commonly found in text style transfer and generation tasks.

MUCOLA, on the other hand, represents a more recent advancement in customizable text generation. It combines the log-likelihood of language models with differentiable constraints into a unified energy function. MUCOLA utilizes a non-autoregressive sampling method based on Langevin dynamics in continous spaces, allowing it to maintain fluency while adhering to user-defined constraints (Kumar et al., 2022). This approach has proven to be a strong baseline in customized text generation but suffers from prompt deviation and mismatches between continuous and discrete representations (Khashabi et al., 2022).

Our work builds upon these concepts by introducing a discrete Langevin dynamics approach that offers a more efficient gradient-based sampling method specifically designed for discrete spaces. This enables robust conditional generation based on desired styles without the need for additional training, positioning our approach as an improvement over both MixMatch and MUCOLA in customized style transfer and text generation tasks.

3 Gradient Based Discrete Sampling on EBMs

The sections provide detailed information about our proposed approach. First, we explain the EBM we will use for sampling. Then, we describe how the discrete sampling approach works with this EBM.

3.1 Energy-Based Model Formulation

We formulate the probability distribution over sequences $\mathcal{S}$ in an EBM as:

$$p(s;\theta) = \frac{\exp(-E(s;\theta))}{\sum_{s' \in \mathcal{S}} \exp(-E(s';\theta))} \quad (1)$$

where $E(s;\theta)$ denotes the energy of sequence s parameterized by θ. Lower energy values correspond to higher probabilities. In our approach to customized generation, we utilize two separate probability distributions over $\mathcal{S}$: one for modeling well-formedness $p_1(s)$ and another for modeling positivity $p_2(s)$ (Mireshghallah and Berg-Kirkpatrick, 2021). A natural solution for generating samples that are both well-formed and positive is to draw from a distribution proportional to the product of these two distributions:

$$p_{\texttt{required}}(s) \propto p_1(s) \cdot p_2(s). \quad (2)$$

Instead of using explicit probability distributions, we assume access to expert blackboxes that provide scalar non-probabilistic energy scores $E_1(s)$ and $E_2(s)$ indicating the fitness of a sequence with respect to well-formedness and positivity, respectively. Under the product of experts framework, the required probability distribution can be expressed as:

$$\log p_{\texttt{required}}(X) = -(E_1(X) + E_2(X)) - \log Z. \quad (3)$$

This shows that the product of expert models results in an energy model where the total energy is the sum of the individual energy scores from the expert models. Inspired by this, the proposed framework for customized generation involves forming linear combinations of various black-box experts to obtain a distribution where the samples meet the desired generation criteria:

$$U(s) = \sum_{i=1}^{k} \alpha_i E_i(s) \quad (4)$$

where k is the number of expert components, and α_i are hyperparameters controlling their influence. For our experiments we use:

1. $E_{\texttt{mlm}}(s)$: We use BERT-based model with an energy parameterization that is the negative sum of unnormalized logits computed iteratively at each position.

2. $E_{\texttt{disc}}(s)$: This expert provides the raw logits of a discriminator for target attributes (task specific classifier). For instance, for positive sentiment, $E_{\texttt{disc}}(s) = -\log p(+|s)$.

3. $E_{\texttt{hamm}}(s; s')$: This represents the Hamming distance between s and a reference sequence s', penalizing token-level deviations, useful for minor edits.

3.2 Discrete Sampling

To sample from the described EBM, we apply a discrete Langevin sampler inspired by Zhang et al. (2022). They introduced a discrete Langevin proposal, analogous to the Langevin algorithm for continuous domains. Sampling from the proposal distribution $q(\cdot|s)$ generates the next position, similar to a Gaussian distribution in continuous spaces but adapted for discrete spaces:

$$q(s'|s) = \frac{\exp\left(-\frac{1}{2\eta}\|s' - s - \frac{\eta}{2}\nabla U(s)\|_2^2\right)}{Z_{\mathcal{S}}(s)} \tag{5}$$

where η is the step size and $Z_{\mathcal{S}}(s)$ is calculated as:

$$Z_{\mathcal{S}}(s) = \sum_{s' \in \mathcal{S}} \exp\left(-\frac{1}{2\eta}\|s' - s - \frac{\eta}{2}\nabla U(s)\|_2^2\right) \tag{6}$$

Although computing $Z_{\mathcal{S}}(s)$ is costly, this proposal can be factorized coordinate-wise, allowing efficient parallel updates:

$$q(s'|s) = \prod_{i=1}^{d} q_i(s_i'|s) \tag{7}$$

where $q_i(s_i'|s)$ is a categorical distribution calculated as:

$$q_i(s_i'|s) = \psi\left(\delta\left(\frac{1}{2}\nabla U(s)_i(s_i' - s_i) - \frac{(s_i' - s_i)^2}{2\eta}\right)\right) \tag{8}$$

where ψ represents categorical distribution and δ denotes softmax function. This factorization ensures that the overall cost depends linearly on sequence length, enabling efficient exploration of the space with gradient information. The proposal is then used with Metropolis-Hastings (MH) step to ensure the Markov chain converges to the target distribution. The MH step accepts the proposed position s' with probability:

$$\min\left(1, \exp(U(s') - U(s))\frac{q(s|s')}{q(s'|s)}\right) \tag{9}$$

3.2.1 Parameterizing Step-Size

A novel contribution of our work is the improvement of the proposal function described by Zhang et al. (2022) by parameterizing the step size. During our experiments, we observed that while the original proposal is effective within local modes, it struggles to escape these modes compared to a random walk sampler. To address this, we modify the proposal function in Equation 8 by parameterizing the step size, enabling a better balance between exploration and exploitation. This modification allows for thorough exploration of current local modes and permits larger steps to escape to better proposals. To achieve this balance, we implement a cyclical schedule for the step size.

$$\eta_k = \max\left(\eta_{\text{max}} \cdot \cos\left(\frac{\pi \mod (k, K)}{K}\right) + 1, \eta_{\text{min}}\right) \tag{10}$$

where η_{max} and η_{min} define the range of step sizes over each cycle, k is the iterator and K defines the total number of sampling steps.

3.3 Token Sampling Limitation

To make our sampling approach more stable, we added a limit on the number of tokens updated in each iteration. The original proposal allowed updating all tokens at once, but this often caused instability. We attribute the instability occurred to the $E_{\text{mlm}}(s)$ function calculated as the negative sum of unnormalized logits computed iteratively at each position, leading to coordinate gradients pulling in conflicting directions. By limiting the token updates to between 3 and 5 per iteration, we achieved better performance stability.

4 Experiments

We apply our proposed approach to style transfer tasks, focusing on sentiment transfer as our primary task. Our method's performance on sentiment transfer is demonstrated using the Yelp dataset test set (Shen et al., 2017; He et al., 2020), which includes 1000 sentences evenly split by sentiment. We conducted the experiment using an NVIDIA 1660 Super GPU. The step size η_{max} was set to 0.07, and η_{min} was set to 0.03. We performed sampling for 150 steps, limiting the token updates to 4 tokens. α_{mlm}, α_{disc} and α_{hamm} is set to 1, 200 and 60 respectively for sentiment transfer. Overall, given a sample text with negative sentiment, the goal is generate text with positive sentiment or vice-versa.

Our setup employs a `bert-base-uncased` MLM for generating proposals. To obtain E_{disc}, we train BERT-based classifiers on the training set of our datasets to use as attribute discriminators. While we could have used any pre-trained attribute classifier from Huggingface for E_{disc}, we reserved those for use as external attribute classifiers for fair evaluation against baselines.

Method	BLEU (ref) ↑	BertScore (src) ↑	Hamming (src) ↓	Int. Clsf. ↑	Ext. Clsf. ↑	Time (sec) ↓
Reference Text	100.00	1.00	5.80	83.70	85.60	-
MUCOLA	20.11	0.95	1.20	84.87	83.22	32.2
MixMatch	19.71	0.95	1.83	94.72	82.85	34.5
Ours	**21.19**	**0.97**	**1.23**	93.12	**85.21**	**28.6**

Table 1: Sentiment transfer performance on Yelp. *(ref)/(src)* denotes metrics measured with respect to reference/source text. *Int. Clsf.* and *Ext. Clsf.* represent internal and external attribute classifier accuracy, respectively. *Hamming* indicates Hamming distance. Arrows (↑ and ↓) specify whether higher or lower values are better for each metric, respectively. We use `textattack/bert-base-uncased-yelp-polarity` as external classifer. The runtime shown is seconds per sample.

Original	**Transferred**
Ever since Joe's has changed hands it's just gotten worse and worse.	Ever since Joe has arrived unanimously it's always so freeing and effective.
We sit down and we got some really slow and lazy service.	We sit down and I love making these sweet and sensitive lashes.
Blue cheese dressing wasn't the best by any means .	Blue cheese dressing was definitely the best by any means.
The associates program is no longer an option.	The associates program is quite welcome an option.

Table 2: Examples of original and transferred sentences for sentiment transfer task

Metrics	Mix Match	MUCOLA	Ours
Grammaticality (↑)	0.80	0.79	**0.85**
Diversity over Unigrams (↑)	0.61	0.57	**0.64**
Diversity over Bigrams (↑)	0.75	0.89	**0.93**
Diversity over Trigrams (↑)	0.80	0.88	**0.93**

Table 3: Comparison of diversity and grammar metrics between our approach and Mix Match. We use `textattack/roberta-base-CoLA` classifer for grammar score.

We compare our proposed approach against two baselines: (1) MUCOLA, which combines the log-likelihood of language models with differentiable constraints into a single energy function, using a non-autoregressive sampling method based on Langevin dynamics for customized text generation; and (2) MixMatch, which utilizes Gibbs sampling to sample from energy-based models.

The results in Table 1 demonstrate that our proposed approach excels in sentiment transfer tasks on the Yelp dataset. Compared to previous approaches, our approach achieves higher BLEU scores, indicating better sequence generation. This is further corroborated by the higher BERTScore, showing that the generated sequences are more similar to the source text in the embedding space. Additionally, the generated text exhibits a lower Hamming distance, signifying fewer changes to the original text. The sentiment classifier results also favor our approach, indicating superior accuracy in converting text to the desired formality level.

Our approach also effectively finds diverse and desired sequences. This is evidenced by the high unigram, bigram, and trigram diversity as well as grammar score shown in Table 3. Furthermore, in terms of inference speed, the sampler is faster than Mix-Match and MUCOLA as seen in Table 1. Overall, our approach demonstrates superior performance, speed, and diversity in generating the desired text. The results of our sampler for transferring negative to positive sentiment on sample text from the Yelp dataset are presented in Table 2. We also present preliminary samples of negative sentiment text generation in A.1. We aim to extend our approach for customized text generation to more recent large language models, such as GPT-4, LLaMA, and Mistral in future work.

5 Conclusion

In conclusion, our discrete Langevin-based proposal offers a highly efficient gradient-based discrete sampler, demonstrating robust conditional generation capabilities without necessitating additional training. By simultaneously adjusting multiple tokens, it effectively navigates the complex discrete distribution of text, resulting in diverse outputs compared to methods that modify a single token per step. Furthermore, this approach provides a versatile, general-purpose sampler that can be tailored to customize text generation across various tasks. The results affirm these benefits, showcasing our approach's superior performance in generating high-quality, diverse text with enhanced efficiency.

References

Ashutosh Baheti, Maarten Sap, Alan Ritter, and Mark Riedl. 2021. Just say no: Analyzing the stance of neural dialogue generation in offensive contexts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4846–4862.

Alexandra Chronopoulou, Matthew Peters, and Jesse Dodge. 2022. Efficient hierarchical domain adaptation for pretrained language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1336–1351, Seattle, United States. Association for Computational Linguistics.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2019. Plug and play language models: A simple approach to controlled text generation. *arXiv preprint arXiv:1912.02164*.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A Smith. 2020. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3356–3369.

Kartik Goyal, Chris Dyer, and Taylor Berg-Kirkpatrick. 2022. Exposing the implicit energy networks behind masked language models via metropolis–hastings. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360.

Junxian He, Xinyi Wang, Graham Neubig, and Taylor Berg-Kirkpatrick. 2020. A probabilistic formulation of unsupervised text style transfer. In *International Conference on Learning Representations*.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Daniel Khashabi, Xinxi Lyu, Sewon Min, Lianhui Qin, Kyle Richardson, Sean Welleck, Hannaneh Hajishirzi, Tushar Khot, Ashish Sabharwal, Sameer Singh, and Yejin Choi. 2022. Prompt waywardness: The curious case of discretized interpretation of continuous prompts. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3631–3643, Seattle, United States. Association for Computational Linguistics.

Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. 2021. Gedi: Generative discriminator guided sequence generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4929–4952.

Sachin Kumar, Biswajit Paria, and Yulia Tsvetkov. 2022. Gradient-based constrained sampling from language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2251–2277, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Florian Mai, Nikolaos Pappas, Ivan Montero, Noah A Smith, and James Henderson. 2020. Plug and play autoencoders for conditional text generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6076–6092.

Fatemehsadat Mireshghallah and Taylor Berg-Kirkpatrick. 2021. Style pooling: Automatic text style obfuscation for improved classification fairness. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2009–2022.

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

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

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6833–6844.

Wei Xu, Alan Ritter, William B Dolan, Ralph Grishman, and Colin Cherry. 2012. Paraphrasing for style. In *Proceedings of COLING 2012*, pages 2899–2914.

Kevin Yang and Dan Klein. 2021. Fudge: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535.

Ruqi Zhang, Xingchao Liu, and Qiang Liu. 2022. A langevin-like sampler for discrete distributions. In *International Conference on Machine Learning*, pages 26375–26396. PMLR.

A Appendix

A.1 Sentiment Based Text Generation Task

Prompt	Negative Sentiment Sentences
The country	The country is unwanted as a part of English Commonwealth countries.
The lake	The lake was near the three multi-strip ruined towers.
The chicken	The chicken was not eaten as a mid-course meal.
The movie	The movie, directed for Zionist film makers, was a waste of energy.
The pizza	The pizza box was useless, with meaningless writing bordering it.
The painting	The painting shows the dead silence of the small city.
The year	In the year of its official opening, spa baths were a failure.
The city	The city was left derelict, and the palace burned up.
The book	The book copyright was criticized by John S. and Patricia S. Champaign.
The horse	The horse was characterized by a foul-lined face with pinched eyes.
The road	The road was again covered with a continuous foul red mist.
Once upon a time	Once upon a time, fans of this movie hated it.

Table 4: Examples of generated sentences with negative sentiment given prompts. Sentences are generated with 12 tokens using the same classifier as in the style transfer task.

We also share preliminary results for text generation to create negative sentiment text from a prompt. The same classifier used in sentiment-based style transfer is applied. The results are shown in Table 4.