PLM-Based Discrete Diffusion Language Models with Entropy-Adaptive Gibbs Sampling

Anonymous ACL submission

Abstract

Recently, discrete diffusion language models have demonstrated promising results in NLP. However, there has been limited research on integrating Pretrained Language Models (PLMs) into discrete diffusion models, resulting in underwhelming performance in downstream NLP generation tasks. This integration is particularly challenging because of the discrepancy between step-wise denoising strategy of diffusion models and single-step mask prediction approach of MLM-based PLMs. In this paper, we introduce Diffusion-EAGS, a novel approach that effectively integrates PLMs with the diffusion models. Furthermore, as it is challenging for PLMs to determine where to apply denoising during the diffusion process, we integrate an entropy tracking module to assist them. Finally, we propose entropy-based noise scheduling in the forward process to improve the effectiveness of entropy-adaptive sampling throughout the generation phase. Experimental results show that Diffusion-EAGS outperforms existing diffusion baselines in downstream generation tasks, achieving high text quality and diversity with precise token-level control. We also show that our model is capable of adapting to bilingual and low-resource settings, which are common in real-world applications.

1 Introduction

As diffusion models significantly enhance the quality and diversity of generated outputs in continuous domains such as images and audio (Song et al., 2021b), recent research has increasingly applied diffusion models to NLP, and demonstrate their promising performance over autoregressive models (Li et al., 2022; Gong et al., 2023a; He et al., 2023; Yuan et al., 2023; Lovelace et al., 2023; Chen et al., 2023; He et al., 2023; Lou et al., 2024; Zhou et al., 2024; Shi et al., 2024; Sahoo et al., 2024; Zheng et al., 2024; Nie et al., 2024). Moreover, these models have been shown to be scalable, and are able to address the challenge of generalizing bidirectional relationships while maintaining sensitivity to the temporal alignment between training and test data, a conventional architectural limitation often observed in autoregressive models (Nie et al., 2024).

Although recent Discrete Diffusion Language Models (DDLMs) have demonstrated high performance in NLP tasks such as unconditional and open-ended generation compared to Continuous Diffusion Language Models (CDLMs) (Lou et al., 2024), existing DDLMs continue to face limitations in conditional generation, thereby restricting their broader applicability across diverse NLP domains. Our experiments suggest that models such as SEDD falls short in such tasks.

To enhance performance of DDLMs, PLMs can be integrated into diffusion models. This approach leverages the capability of PLMs to elicit improved generalization capabilities in unseen tasks while simultaneously exploiting the controllability and diversity strengths inherent to diffusion models. However, integrating PLMs into diffusion models is non-trivial as PLMs typically predict masked elements in a single step, whereas diffusion models require step-wise denoising based on the overall semantics of each timestep sequence, and such gap yields limited results. Therefore, we need to consider such inconsistencies to effectively adopt PLMs into DDLMs by a new methodology.

In this paper, we introduce **Diffusion-EAGS**, a novel approach that effectively integrates Mask Language Model (MLM)-based PLMs with DDLMs for conditional generation. To address the gap between the step-wise nature of diffusion models and the one-step prediction strategy of PLMs, we begin by adopting PLMs as the denoising function at each step of the diffusion process. Since identifying where to apply denoising during the diffusion process is challenging for PLMs in our experiments, we incorporate an entropy tracking

Type \ Dataset	OpenWebText (Gokaslan and Cohen, 2019)	RocStories (Mostafazadeh et al., 2016)	Deontology (Hendrycks et al., 2023)	Question Generation (Dhingra et al., 2017)	QQP (Wang et al., 2017)	ALMA (Xu et al., 2024a)	ParaDetox (Logacheva et al., 2022)
Open-ended Generation	✓	✓	△	✓	×	×	×
Conditional Generation	×	✓	✓	✓	✓	✓	✓
– Context Provided ?	-	✓	✓	✓	✓	✓	✓
– Content Provided ?	-	×	△	✓	✓	✓	✓
– Format Provided ?	-	-	×	×	×	✓	✓

Table 1: Each dataset has a different level of conditional contraints even if they are conditional generation tasks. ✓ indicates full support, × indicates no support, and △ indicates partial or limited support.

module to support their operation based on entropy information. To train DDLMs to effectively exploit an entropy tracking module, we propose an entropy-based noising scheduling during training phase in a forward process. Specifically, our entropy-based noising scheduling noises lower entropy tokens first, thereby learning to progressively generate sequences guided by the entropy from entropy tracking module.

Experimental results demonstrate that Diffusion-EAGS achieves outstanding performance compared to baseline models across various conditional generation tasks. Furthermore, our model exhibits higher diversity in certain tasks compared to LLMs and can facilitate token-level generation, and validate that our model can also adapt to both bilingual and low-resource settings, which are frequently encountered in practical applications, indicating the potential applicability of our model across a wide range of conditional generation tasks.

2 Task Setting

2.1 Fine-Grained Conditional Generation

In conditional generation tasks, the level of conditional constraint imposed by the dataset plays a critical role in shaping the generation process. As shown in Table 1, conditional constraints are diverse across datasets. In our task, we categorize these constraints into three levels: (1) the provision of context alone, requiring the continuity of the prefix; (2) the provision of specific content to be included in the target sequence, necessitating the inclusion of certain keywords; and (3) the provision of semantic content formatting, such as transforming toxic sentences into safer alternatives or converting text from the source language to a target language. In our study, we aim to develop a universal diffusion framework capable of being applied across a wide range of conditional generation tasks.

2.2 DDLM with BERT

BERT as Markov Random Field (MRF) As the pretraining paradigm has significantly contributed to the success of language models, the integration of PLMs is essential in the development of language models. Nevertheless, adopting PLMs to DDLMs is a complex task, as DDLMs require a step-wise denoising strategy, while PLMs are trained to predict masked elements in a single step. To effectively integrate PLMs into DDLMs, it is necessary to resolve such a discrepancy. Therefore, we begin by interpreting the intrinsic characteristics of MLMs as Markov Random Field (MRF), as demonstrated in Wang and Cho (2019). Details are in Appendix B.

constrained MRF (cMRF) Even though open-ended MLMs are MRFs, we observe a significant increase in log-potential values for sequences when guided by the RocStories conditions, as shown in Figure 1. Additional experiments supporting this observation are detailed in Appendix A.

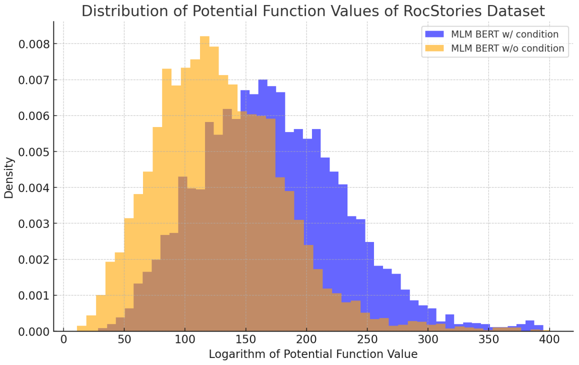

Figure 1: When a condition is provided, the distribution of potential values for the samples is shifted on a logarithmic scale.

This suggests that conditional generation models differ from open-ended models in randomness, making it crucial to investigate whether conditionally constrained spaces adhere to MRF properties. Through the derivation process outlined in Appendix B, we demonstrate that these spaces retain the characteristics of an MRF.

Given a sequence of fully connected random variables $X = (x_1, x_2, \ldots, x_L)$ with a set of observed variables $Y = (y_1, y_2, \ldots, y_K)$, $f_\theta(X \backslash \{x_l\}; Y)$ can be represented by logits from MLM given the sequence $X \backslash \{x_l\}$ and the observed variables Y. Then, log-potential can be calculated as:

$$\log \phi_l(X;Y) = \begin{cases} 1h(x_l)^T f_\theta(X \backslash \{x_l\}; Y), & \text{if [MASK]} \in X_{l-1:l+1}, \\ 0, & \text{otherwise} \end{cases}$$

where $1h(x_l)$ is a one-hot vector with x_l set to 1. By this, we can define the potential function for the clique as:

$$\phi(X;Y) = \exp\left(\sum_{l=1}^{L} \log \phi_l(X;Y)\right) \qquad (1)$$

Then, the constrained probability of sequence X is:

$$p_\theta(X;Y) = \frac{1}{Z(Y,\theta)} \prod_{l=1}^{L} \phi_l(X;Y)$$

Using the softmax function, the distribution of a variable x_l given the observed variables Y is:

$$p(x_l|X\backslash\{x_l\},Y) = \frac{\exp(1h(x_l)^T f_\theta(X\backslash\{x_l\};Y))}{\sum_{m=1}^{M}\exp(1h(m)^T f_\theta(X\backslash\{x_l\};Y))} \qquad (2)$$

This formulation implies that the dataset-constrained generation process in BERT can still be interpreted as a MRF with constraint Y, where the log-potential functions can be derived from the output logits of MLM.

3 Methodology

In this section, we propose Diffusion-EAGS that integrates PLM with DDLMs with Entropy-based Adaptive Gibbs Sampling (EAGS) and Entropy-based Noise Scheduling (ENS). The overview of our process is in Figure 2.

3.1 Generation Process: Entropy-based Adaptive Gibbs Sampling (EAGS)

As discussed in Section 2.2, MLM-based PLMs can be interpreted as cMRFs, allowing Gibbs sampling to be applied at each denoising step in DDLMs by leveraging the properties of MRFs. As PLMs often struggle to select the next denoising targets, it is important to highlight that the primary purpose of adopting Gibbs sampling is not only to bridge the gap between PLMs and diffusion models but also to assist PLMs in identifying which absorbing tokens should be denoised. Therefore, we propose Entropy-Adaptive Gibbs Sampling (EAGS) to address the challenge PLMs face in identifying the appropriate denoising targets. In EAGS, the entropy of each variable quantifies the amount of uncertainty at a specific position within the sequence. This enables sampling of each token x_l from its constrained distribution $p(x_l|X\backslash\{x_l\};Y)$ in the descending order of entropy. By prioritizing the generation of the least informative parts of the sequence, EAGS facilitates the creation of more structured sequences by leveraging the syntactic context that has already been established.

Specifically, according to Equation (2), the probability $p^t(x_l = v_m|Y, f_\theta)$ of the token v_m given the context Y and f_θ at step t can be calculated as:

$$p^t(x_l = v_m|Y, f_\theta) = \frac{\exp(f_\theta(v_m|X^t))}{\sum_{m'=1}^{M}\exp(f_\theta(v_{m'}|X^t))}$$

, where $f(v_m|X^t)$ is the logit corresponding to the token v_m for the position x_l at step t. That is, we can compute the conditional probability for each masked position. With the conditional probability, the entropy H for a variable x_l, logits function f_θ, and the dataset context Y is derived as:

$$H(x_l|Y, f_\theta) = \\ -\sum_{m=1}^{M} p(x_l = v_m|x_l, Y, f_\theta)\log p(x_l = v_m|x_l, Y, f_\theta)$$

With H, our approach for T step-generation process can be formalized as follows:

1. **Entropy Calculation**: Compute the entropy $H(x_l \mid Y, f_\theta)$ for each variable x_l.
2. **Variable Selection**: Select the variable x_{l^*} with the highest entropy for sampling

   $$l^* = \arg\max_l H(x_l \mid Y, f_\theta)$$
3. **Sampling**: Sample x_{l^*} from its conditional distribution $p(x_{l^*} \mid Y, f_\theta^*)$.
4. **Update**: Update the conditional distributions and entropy for the affected variables.
5. **Iteration**: Repeat Steps 1 through 4 until t=T.

3.2 Forward Process: Entropy-based Noise Scheduling (ENS)

To enhance the facilitation of Adaptive Gibbs Sampling in the generation process, it is essential to simulate a similar process in the training phase. Therefore, we schedule the forward process of diffusion training based on the entropy $H(x_l)$ of position x_l with the input sequence $[Y|X]$. Specifically, the position-wise entropy is calculated by the PLM that shares a similar language base with diffusion-trained model. Assuming the diffusion process progresses over T steps, at each step t, we mask the L/T number of positions with the lowest entropy from the set $\{x_1, \ldots, x_L\}$. The masking process at step t in position l is described by the denoising matrix Q_{tl}.

$$Q_{tl} = \begin{bmatrix} q_{11} & 0 & \cdots & 0 & q_{1,M} \\ 0 & q_{22} & \cdots & 0 & q_{2,M} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & q_{M-1,M-1} & q_{M-1,M} \\ 0 & 0 & \cdots & 0 & q_{MM} \end{bmatrix}$$

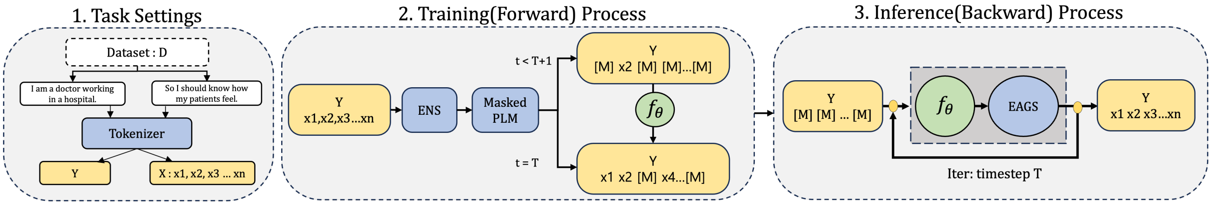

Figure 2: Given a conditional constraint Y and a generation target X, the training process utilizes Entropy-based Sampling (ENS) to determine which positions in the masked sequence should be denoised when transitioning from one timestep to the next. The tokens at the corresponding masked positions ([M] denotes masked tokens) are then generated using f_θ. The loss is subsequently computed, and the parameters are updated accordingly. During the inference process, the sequence X, initially composed of mask tokens, is iteratively refined through the diffusion process to generate X that satisfies the given condition Y.

, where

$$q_{mn} = \begin{cases} q_{mm} = 1 & \text{if } x_l \notin \text{MIN}([H(x_1), \cdots, H(x_L)], \frac{L}{T}) \\ q_{mM} = 1 & \text{if } x_l \in \text{MIN}([H(x_1), \cdots, H(x_L)], \frac{L}{T}) \\ 0 & \text{otherwise} \end{cases}$$

This entropy-based noise scheduling approach ensures that the forward process in diffusion training closely mirrors the generation process, thereby enhancing the effectiveness of EAGS in language generation.

3.3 Training and Inference

Algorithm 1: ENS and EAGS Algorithm

```
ENS Process:
Input: Context $Y$ and Dataset $D$
for Batch Step $= 0$ to $N$ do
    $x \sim D$                                   // Sample data from D
    $t \sim \text{Randint}(0, T)$                 // Sample random timestep
    $f \leftarrow \text{PLM}(x \mid Y)$           // Compute logits using the PLM
    $\mathcal{H} \leftarrow H(x \mid Y, f_\theta)$   // Calculate Entropy
    $x^t \leftarrow \text{Forward}(x_0, \mathcal{H}, t)$   // Forward at t
    $x^{t+1} \leftarrow \text{Forward}(x_0, \mathcal{H}, t+1)$   // Forward at t + 1
    $L_s = -\sum_i q(x_i^t \mid x_{t+1}) \log p_\theta(x_i^t \mid x_{t+1})$
                                                 // Loss calculation
end

EAGS Process:
Input: Sequence Length $L$, Total Timestep $T$,
  Trained Model $M$, Mask Sequence Generator $G_M$,
  and Context $Y$
for $t = T$ to $0$ do
    if $t = T$ then
        $x^T \leftarrow G_M(L, Y)$               // Initialize a sequence of L
    else
        $f \leftarrow M(x^t, Y)$                 // Compute logits at timestep t
        $l^* \leftarrow \arg\max_l H(x_l^t \mid Y, f_\theta)$
                                                 // Obtain nth largest entropy tokens
        $x^{t-1} \leftarrow \text{Sampling}(x^t, l^*, Y)$
                                                 // Sample from the previous timestep
    end
end
```

Our whole process is in Algorithm 1. Distinct from the prevailing methodologies in diffusion models as described by Ho et al. (2020a) and Austin et al. (2023), we do not employ the PLM parameterization strategy $\widetilde{p}_\theta(\widetilde{z}_0|z_t, t)$, which preserves the original semantic embedding spaces during the training phase. We empirically find that PLM parameterization restricts the diversity of generated responses. Instead, to ensure the completeness of sentences, we implement Minimum Bayes Risk (MBR) decoding in the final generation step. We follow the traditional diffusion loss in Equation (Ho et al., 2020b), changing MSE with Cross Entrpy Loss.

4 Experiments

4.1 Tasks and Datasets

Social Datasets The Paradetox dataset (Logacheva et al., 2022) focuses on removing profanities. The Deontology from ETHICS dataset (Hendrycks et al., 2023) evaluates ethical judgment based on scenarios. **Paraphrase** The QQP dataset (Wang et al., 2017) assesses paraphrase detection by treating one question as a paraphrase of another. **Question Generation** We use the Quasar-T dataset (Dhingra et al., 2017) to generate questions from passages and answers. **Open-ended Generation** The RocStories (Mostafazadeh et al., 2016) dataset is used to generate coherent story continuations from a given context. Detailed explanations are available in Appendix C and Dataset statistics are shown in Table 8.

4.2 Baselines

Since our objective is centered on generation tasks, we compare Diffusion-EAGS with four categories of baselines with the model similar to the size of *RoBERTa-base* (Liu et al., 2020): Auto-regressive Models (ARMs), CMLMs (Conditional Masked Language Models, Ghazvininejad et al. (2019a)), CDLMs, and DDLMs. For **ARMs** (Vaswani et al., 2023), we employ GPT-2 (Radford et al., 2019), which is renowned for its proficiency in generation tasks. For **CMLMs**, we utilize CMLM (Ghazvininejad et al., 2019a) and DisCo (Kasai et al., 2020), which are transformer-based NAR models. Inference is carried out using the Mask-Predict (Ghazvininejad et al., 2019a) and the Easy-First (Kasai et al., 2020) inference algorithm respectively. For **CDLMs**, our baseline includes DiffuSeq (Gong et al., 2023a) and DINOISER (Ye et al., 2024). DiffuSeq is a diffusion model specifically designed for sequence-to-sequence applications, whereas DINOISER adaptively determines the range of sampled noise scales during training. For **DDLMs**, we utilize DiffusionBERT (He et al., 2022) , a state-of-the-art model in the research series of DiffusionLM (Li et al., 2022), and SEDD (Lou et al., 2024), a state-of-the-art model of open-ended generation in diffusion language generation domain. For SEDD, we download the pretrained version and fine-tune on it. Experimental results on LLMs can be found in Appendix E.

4.3 Metrics

Quality metrics To measure the quality of the generated texts, we use Perplexity based-on GPT-2 Large and GPT-2 XL, SOME (Yoshimura et al., 2020), grammar metric based on the neural net, LLM-c (Lin and Chen, 2023) to measure the plausibility of the narratives, LLM-t (Koh et al., 2024) to measure toxicity, and MAUVE (Pillutla et al., 2021), measuring the gap between text distributions despite divergence.

Diversity Metrics Traditional diversity metrics Self-BLEU (Zhu et al., 2018) and distinct-n (Li et al., 2015) are employed to evaluate the generated texts. We also adopt Vendi Score (VS) (Friedman and Dieng, 2023), an interpretable diversity metric, which quantifies the effective number of unique samples in a given set. Both the n-gram and embedding variations are utilized, where embedding VS interprets diversity of semantics.

4.4 Experimental Details

We employ roberta-base as encoder-based PLM. The learning rate is set at 5e-4, and a naive categorical sampling strategy is adopted. Adapted to data statistics, the maximum lengths for QG, QQP, and Paradetox is set to 64, while for Deontology set to 48. Furthermore, in accordance with the minimum construction length, the number of steps is configured to 5 and 20 size of MBR. We use 1 A100 GPU, and the batch size is set to 256.

5 Results

As shown in Table 3, 2, 4, our model consistently exhibits exceptional performance in terms of text quality while simultaneously maintaining diversity when compared to baseline models. As illustrated in Table 3, Diffusion-EAGS generates the responses with the highest PPL score for QG, and highest MAUVE and PPL score for QQP.

In Table 2 Paradetox, our model demonstrates superior performance across all evaluated metrics. Such phenomenon represents that our model based on encoder-based PLMs show robustness on diverse perturbations from daily dialogues. When PPL exceeds 600, the model is considered to have failed in generating natural sequences, and is thus represented in gray color. In the context of Deontology, our model exceeds the baseline PPL and MAUVE scores, whereas SOME score represent the sufficient quality of text with the highest diversity score of 4.755. Additionally, in Table 4 where semantic consistency is crucial, our model significantly outperforms the original dataset's PPL by 23 points by leveraging PLMs while maintaining a high diversity score of 4.837. In the paradetox experiment, the text quality produced by the CMLM and SEDD models was found to be low. Consequently, these models were excluded from subsequent experiments.

Diffusion-EAGS demonstrates high level of text quality surpassing that of GPT-2 in Table 3 in text quality, and the highest MAUVE score of 0.733 in Table 2-ParaDetox. ParaDetox is colloquial dataset including slang, numerous abbreviations, and various perturbations, so our model demonstrate robustness to such perturbations with 69.5 PPL. As for diversity, our model consistently outperforms GPT models in VS(ngram) and VS(emb) in Table 3, 2, and 4. These results underscore our model's efficacy in generating diverse responses.

Notably, CDLMs demonstrate a noticeable deficiency in diversity. In contrast, our model excels at producing significantly more diverse sequences. Furthermore, our models require only a few steps, while resulting in higher quality and diversity.

DiffusionBERT exhibits limitations in text quality, as shown in Table 3, 2, 4. On the other hand, our model achieves significantly higher scores across all quality metrics. This observation suggests that the naive application of BERT (Devlin

					ParaDetox				
Model	Step	PPL ↓	MAUVE ↑	SOME ↑	VS(ngram) ↑	VS(emb) ↑	self-bleu ↓	distinct-1 ↑	distinct-2 ↑
GPT-2	1	389.1	0.503	0.717	3.925	2.640	0.429	0.312	0.748
CMLM w/ Mask-Predict	10	669.9	0.0234	0.588	1.000	1.000	1.000	0.451	0.633
DisCo w/ Easy-First	10	716.1	0.0344	0.576	1.000	1.000	1.000	0.438	0.583
DiffusionBert	2000	775.9	0.737	0.716	3.101	2.058	0.599	0.424	0.826
DiffuSeq	2000	$\geq 1k$	0.683	0.703	2.059	1.465	0.841	0.410	0.820
DINOISER	20	124.8	0.255	0.767	2.287	2.174	0.981	0.211	0.486
SEDD	1024	$\geq 1k$	NA	0.664	4.746	4.063	0.119	0.451	0.846
Diffusion-EAGS	5	**69.5**	**0.773**	**0.796**	**4.755**	3.659	**0.126**	**0.475**	**0.834**
					Deontology				
	Step	PPL ↓	MAUVE ↑	SOME ↑	VS(ngram) ↑	VS(emb) ↑	self-bleu ↓	distinct-1 ↑	distinct-2 ↑
GPT-2	1	92.0	0.131	**0.860**	3.665	3.126	0.425	**0.474**	**0.874**
DiffuSeq	2000	352.8	0.005	0.703	2.273	1.915	0.753	0.267	0.745
DINOISER	20	131.3	0.008	0.740	2.287	2.174	0.824	0.309	0.713
DiffusionBert	2000	295.5	0.306	0.787	4.258	3.458	0.229	0.445	0.849
Diffusion-EAGS	5	**55.1**	**0.412**	0.835	**4.898**	**4.009**	**0.056**	0.418	0.806

Table 2: Social Generation – Diversity values associated with higher perplexity (PPL) are displayed in gray, as increased perplexity typically indicates degenerate sequences.

					QQP				
Model	Step	PPL ↓	MAUVE ↑	SOME ↑	VS(ngram) ↑	VS(emb) ↑	self-bleu ↓	distinct-1 ↑	distinct-2 ↑
GPT-2	1	66.270	0.112	0.754	3.886	2.566	0.423	0.344	0.787
DiffuSeq	2000	124.247	0.00674	0.709	1.927	1.242	0.813	0.226	0.543
DINOISER	20	79.742	0.0042	0.821	1.421	1.126	0.935	0.264	0.542
DiffusionBert	2000	500.959	0.0709	0.618	**4.489**	**2.836**	**0.196**	0.321	0.761
Diffusion-EAGS	5	**48.106**	**0.683**	**0.824**	4.006	2.390	0.338	**0.421**	**0.832**
					QG				
			Text Quality				Diversity		
Model	Step	PPL ↓	MAUVE ↑	SOME ↑	VS(ngram) ↑	VS(emb) ↑	self-bleu ↓	distinct-1 ↑	distinct-2 ↑
GPT-2	1	124.8	0.141	0.759	4.564	3.130	0.176	0.210	0.629
DiffuSeq	20	395.0	0.149	0.730	1.555	1.274	0.901	0.170	0.564
DINOISER	2000	155.9	**0.159**	0.776	1.396	1.121	0.944	0.166	0.553
DiffusionBert	2000	513.6	0.150	0.712	3.040	2.209	0.566	**0.392**	0.759
Diffusion-EAGS	5	**80.7**	0.121	**0.782**	**4.646**	**3.538**	**0.152**	**0.403**	**0.798**

Table 3: QG & QQP Generation

Model	PPL ↓	SOME ↑	LLM-c ↑	VS(ngram) ↑	self-bleu ↓
Original Data	100.6	0.895	1		
GPT-2	88.5	**0.856**	**0.88**	4.722	0.124
DiffusionBert	318.2	0.783	0.72	4.735	0.088
SEDD	273.2	0.827	0.59	**4.859**	**0.044**
Diffusion-EAGS	**67.3**	0.844	0.87	4.837	0.058

Table 4: ROC Generation – Because ROC is open-ended generation task, we include SEDD in this experiment.

et al., 2019) within the diffusion process fails to fully harness the capabilities of PLMs. SEDD, an open-ended generation model shows low performance under text generation in highly constrained tasks such as ParaDetox. Additionally, our model does not require more than 5 diffusion steps, as it can adaptively recover the absorbing state through an entropy-based generation approach. A detailed comparison is provided in Appendix D.

6 Analysis

6.1 Ablation Study

To explore the effectiveness of our model's components, we conduct ablation studies focusing on three key elements: Entropy Adaptation, stepwise Gibbs Sampling, and PLM in Table 5.

The omission of EAGS initially leads to a substantial decline in performance and text quality, *producing degenerated results similar to those of traditional CMLMs*.

We find that EAGS significantly contributes to a gradual entropy decrease, as shown in Appendix J. These observations highlight the critical role of selecting the sampling position based on given information in sequence generation. Additionally, excluding the use of the diffusion generation process, without the stepwise sampling, leads to a drastic reduction in overall performance, with PPL increasing by more than 1000 points and diversity scores dropping below 2. Consequently, our cMRF approach, integrated with the diffusion model, proves indispensable in aiding the PLM in effectively generating sequences. Furthermore, it is evident that without the incorporation of PLMs, the generation of natural sequences is unachievable.

	Dataset	PPL	MAUVE	SOME	VS(ngram)	VS(emb)
Diffusion-EAGS	Deont	55.1	0.412	0.835	4.898	4.009
	Roc	67.3	0.87	0.844	4.837	3.999
w/o EAGS	Deont	667.9	0.022	0.617	4.767	3.928
	Roc	1084.9	0.035	0.613	4.874	3.957
w/o Gibbs Sampling	Deont	1426.7	0.011	0.584	2.378	1.923
	Roc	1293.1	0.010	0.534	1.531	1.338
w/o PLM	Deont	≥2K	0.005	0.645	4.758	3.402
	Roc	≥2K	0.004	0.604	4.315	2.994

Table 5: Ablation Study

6.2 Case Study: Keyword Based Generation

Our model operating within discrete space enables us to manipulate the output sequences using explicit instructions. To further explore this capability, we conduct the generation of sequences based on keywords positioned in the middle and at the end of masked sequences, which is challenging for AR models (Keskar et al., 2019). Initially, we provide the same contextual input while varying the keywords. In the masked states, we randomly select positions, replacing them with the specified keywords. The results in Table 7 demonstrate that the generated sequences seamlessly integrate the keywords with context-specific semantics.

These findings show that our model can effectively integrate specific conditional information in an interpretable manner. Experimental results also indicates its substantial potential and suitability for diverse applications where direct controllability is crucial, such as in story generation.

6.3 Bilinguality & Low Resource Settings

Model	SacreBLEU	COMET	XCOMET
DisCo			
w/ Easy-First	3.2806	0.2447	0.2414
w/ Mask-Predict	3.2862	0.2444	0.2414
DisCo-m			
w/ Easy-First	3.7423	0.2468	0.2122
w/ Mask-Predict	3.7748	0.2466	0.2119
Diffuseq-v2	1.90	0.3242	0.2628
SEDD			
w/ from scratch	0.14	0.2375	0.2035
w/ pretrained	0.25	0.2504	0.2076
DiffusionEAGS-NLLB	**20.9297**	0.5720	0.6629
NLLB-naive-600M	4.1827	0.6134	0.7818
mBART-50-FT	19.6536	**0.7576**	**0.8748**

Table 6: En-De Translation Results

Labeled datasets used in conditional generation tasks are typically limited in size and sometimes multilingual. To further assess our model's performance in conditional generation, particularly in terms of language extension and resource scarcity, we conduct additional experiments on a translation task. Specifically, we utilize the 18k *en*↔*de* human-curated dataset by Xu et al. (2024a,b). For our model, we employ a pretrained NLLB (Costa-jussà et al., 2022) as a non-autoregressive (NAR) approach for controlling language output separately. This approach is selected due to the difficulty of controlling token generation in a small-scale multilingual BERT, which suffers from interference issues (Shaham et al., 2023).

Although our baseline diffusion models demonstrated significant performance improvements over CMLMs, CMLMs were primarily developed to enhance translation performance. Therefore, we conducted additional experiments to further validate our methodology such as Mask-and-Predict and Easy-First, diffusion models such as Diffuseq-v2 (Gong et al., 2023b) and SEDD, traditional translation models such as mBART-50 (Tang et al., 2020) and NLLB. For evaluation metrics, we utilize sacreBLEU (Post, 2018) and neural-net scores such as COMET (Rei et al., 2020) and XCOMET (Guerreiro et al., 2023).

Table 6 shows that predicting the target sequence without leveraging a multilingual PLM proves to be challenging. All diffusion baseline models struggle to produce correct outputs. For example, the pretrained SEDD model fails to effectively leverage conditional information, even after finetuning on the training datasets, consistent with the limitation observed in Section D.3. Similar challenges arise in NAR transformer baselines. Despite constructing the vocabulary using the pretrained mBART-50 model (DisCo-m), the underlying issues remain. On the other hand, our proposed model, by incorporating a PLM, demonstrates promising results. In addition, these results show that our methodology can also be used with encoder-decoder models.

Interestingly, the output of the pretrained NLLB model (NLLB-naive-600M, not finetuned) reveal that neural network-based metrics are susceptible to the interference problem, specifically translated by other languages, even though we provide the language specific token. While such issues result in lower BLEU scores, COMET and XCOMET often interpret them as semantically coherent, indicating a potential direction for future work to improve translation evaluation metrics. Despite these phenomena, a performance gap between translation models and DDLM remains. This suggests that future research should address the semantic capabilities of diffusion models to help bridge this gap. Details are in Appendix I.

6.4 Diversity Saturation

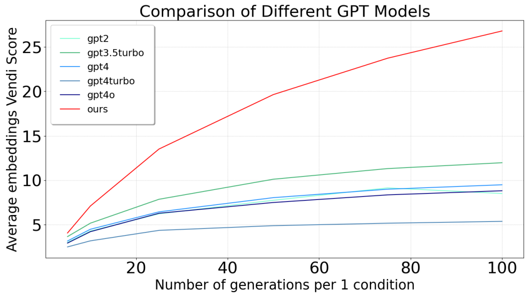

Figure 3: Diversity graph with increasing generation numbers in 'Deontology' dataset

Inspired by the observation that Diffusion-EAGS consistently excel in terms of diversity across all results, we delve further into the diversity capabilities

Context		Jake was playing with his toys. He accidentally broke his favorite one. He cried a lot over it. His parents decided to replace it for him.
Keyword	not stop	Jake just could **not stop** crying.
	Jake feel	It made **Jake feel** So much better.
	would enjoy	Jake said he **would enjoy** the new toy
Context		Neil was in Sofia, Bulgaria. He was enjoying a trip backpacking through Europe. ... He thought the food and culture in Sofia were the best.
Keyword	Bulgaria!	Things were looking great in **Bulgaria!**
Context		Karen wanted to go on a trip to France. She started doing research on the trip. She decided to book a week long trip. She left the next day for her trip.
Keyword	her trip	She spent almost a week there during **her trip**.

Table 7: Keyword-based Generation

of our model. We assess the diversity performance in conditional generation compared to LLMs. We measure the VS for 5 to 100 generations under a single condition. Such experiment demonstrates the extent to which the model's output diversity saturates, enabling a comparison of asymptotic diversity performance. The experiment is conducted on the 'deontology' dataset which allows high output diversity in its settings. Details of using LLMs are provided in Appendix G.

Figure 3 demonstrates that the diversity saturation graph for Diffusion-EAGS has a relatively steep slope, while GPT models saturate at lower values. The embedding VS of all GPT series saturates below 13. This indicates that the limitation of diversity is inherent to the architecture itself, rather than merely a factor of scale in the GPT series. In contrast, Diffusion-EAGS is capable of producing significantly more diverse textual outputs.

7 Related Works

Efforts to integrate generative flow models into sequence generation exploit the distribution shift from a source language to a target language through a series of invertible linear transformations (Ma et al., 2019; Zhang et al., 2024). However, as DDPM (Ho et al., 2020a) demonstrate the effectiveness of generating images, diffusion models have been a major topic of interest within the field of generative flow models (Song et al., 2021a,b). To apply such diffusion methodologies to NLP, there are two main streams - continuous diffusion models and discrete diffusion models.

Continuous diffusion models Diffusion-LM (Li et al., 2022) propose a method of mapping tokenized sequences to embedding dimensions guided by a pretrained classifier. DiffuSeq-v1, v2 (Gong et al., 2023a,b) apply partial noising techniques. The core of these diffusion methodologies lies in the addition and restoration of random noise to facilitate generation. However, authors of Cold Diffusion (Bansal et al., 2022) argue that such operations do not necessarily have to be governed by stochastic randomness.

Discrete diffusion models Building on this rationale, D3PM (Austin et al., 2023) propose the discrete restoration-generation approach and DiffusionBERT (He et al., 2022) adopt PLMs to DDLM. SEDD(Lou et al., 2024) propose score entropy inspired by MLM loss. Recent works by Shi et al. (2024) and Sahoo et al. (2024) extend this idea to propose a simplified view of the discrete framework, and obtain better empirical results. Zheng et al. (2024) further close the gap between discrete diffusion models and masked models, and also, by correcting the numerical precision error during the sampling process in SEDD-based models, reveal that the true generation perplexity of DDLMs lags behind that of AR models. These research make an improvement on the open ended generation task. Furthermore, Venkatraman et al. (2024) showcase that SEDD priors can be used to sample from a posterior distribution, namely text infilling, and Nie et al. (2024) demonstrate that DDLMs are scalable, and have advantages over traditional AR models.

8 Conclusions

In this work, we introduce Diffusion-EAGS, integrating PLMs with diffusion models for conditional generation. By leveraging step-wise diffusion models and the property of MLM, we demonstrate that MLM-based PLMs can serve as denoising function with Entropy-based Adaptive Gibbs Sampling. We also propose entropy-based noise scheduling during training for adaptive sampling. Experimental results demonstrate that Diffusion-EAGS outperforms existing baselines, yielding improved text generation quality, enhanced diversity, broad applicability, and more precise token-level control.

Limitations

While Diffusion-EAGS demonstrates significant improvements in conditional generation tasks, there are several limitations. Firstly, as our method is currently focused on text generation tasks, its applicability to text classification tasks, such as Named Entity Recognition and Part-of-Speech Tagging, remains unexplored. Future research could explore extending this method to other NLP tasks. Secondly, future works could investigate the potential in reducing computational costs through more efficient entropy calculation techniques. Lastly, although our current efforts concentrate on developing and validating our framework using encoder-only and encoder-decoder architectures, the potential integration of autoregressive (AR) models remains unexplored. However, since the denoising function—with its entropy-based, sequential ordering—aligns well with autoregressive decoding strategies, we posit that AR models can also serve as effective initializations within our proposed framework.

Ethical Statements

A notable advantage of Diffusion-EAGS is its efficient training process, which optimizes computational resources with high performance. This efficiency not only reduces the environmental impact of training PLMs but also makes advanced NLP technologies more accessible. Additionally, we have focused on output control and interpretability through the use of a discrete diffusion model. Consequently, our methodology contributes to the effective control of generated outputs in the future. Diffusion-EAGS's strength in tasks involving conditional generation proves beneficial in the field of text generation following social guidelines, where outputs need to strongly depend on the conditions. Specifically, its high diversity performance enables active utilization in dynamically adapting text generations to meet specific social criteria.

References

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. 2023. Structured denoising diffusion models in discrete state-spaces.

Arpit Bansal, Eitan Borgnia, Hong-Min Chu, Jie S. Li, Hamid Kazemi, Furong Huang, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2022. Cold diffusion: Inverting arbitrary image transforms without noise.

Jiaao Chen, Aston Zhang, Mu Li, Alex Smola, and Diyi Yang. 2023. A cheaper and better diffusion language model with soft-masked noise.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Ameet Deshpande, Vishvak Murahari, Tanmay Rajpurohit, Ashwin Kalyan, and Karthik Narasimhan. 2023. Toxicity in chatgpt: Analyzing persona-assigned language models. *arXiv preprint arXiv:2304.05335*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Bhuwan Dhingra, Kathryn Mazaitis, and William W. Cohen. 2017. Quasar: Datasets for question answering by search and reading.

Dan Friedman and Adji Bousso Dieng. 2023. The vendi score: A diversity evaluation metric for machine learning.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019a. Mask-predict: Parallel decoding of conditional masked language models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6112–6121, Hong Kong, China. Association for Computational Linguistics.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019b. Mask-predict: Parallel decoding of conditional masked language models. *arXiv preprint arXiv:1904.09324*.

Aaron Gokaslan and Vanya Cohen. 2019. Openwebtext corpus. `http://Skylion007.github.io/OpenWebTextCorpus`.

Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and Lingpeng Kong. 2023a. Diffuseq: Sequence to sequence text generation with diffusion models.

Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and Lingpeng Kong. 2023b. Diffuseq-v2: Bridging discrete and continuous text spaces for accelerated seq2seq diffusion models.

Nuno M Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André FT Martins. 2023. xcomet: Transparent machine translation evaluation through fine-grained error detection. *arXiv preprint arXiv:2310.10482*.

Zhengfu He, Tianxiang Sun, Qiong Tang, Kuanning Wang, Xuanjing Huang, and Xipeng Qiu. 2023. DiffusionBERT: Improving generative masked language models with diffusion models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4521–4534, Toronto, Canada. Association for Computational Linguistics.

Zhengfu He, Tianxiang Sun, Kuanning Wang, Xuanjing Huang, and Xipeng Qiu. 2022. Diffusionbert: Improving generative masked language models with diffusion models.

Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob Steinhardt. 2023. Aligning ai with shared human values.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020a. Denoising diffusion probabilistic models.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020b. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc.

Jungo Kasai, James Cross, Marjan Ghazvininejad, and Jiatao Gu. 2020. Non-autoregressive machine translation with disentangled context transformer. In *International conference on machine learning*, pages 5144–5155. PMLR.

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

Hyukhun Koh, Dohyung Kim, Minwoo Lee, and Kyomin Jung. 2024. Can llms recognize toxicity? definition-based toxicity metric.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2015. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*.

Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. 2022. Diffusion-lm improves controllable text generation.

Yen-Ting Lin and Yun-Nung Chen. 2023. LLM-eval: Unified multi-dimensional automatic evaluation for open-domain conversations with large language models. In *Proceedings of the 5th Workshop on NLP for Conversational AI (NLP4ConvAI 2023)*, pages 47–58, Toronto, Canada. Association for Computational Linguistics.

Y Liu. 2020. Multilingual denoising pre-training for neural machine translation. *arXiv preprint arXiv:2001.08210*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Ro{bert}a: A robustly optimized {bert} pretraining approach.

Varvara Logacheva, Daryna Dementieva, Sergey Ustyantsev, Daniil Moskovskiy, David Dale, Irina Krotova, Nikita Semenov, and Alexander Panchenko. 2022. ParaDetox: Detoxification with parallel data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6804–6818, Dublin, Ireland. Association for Computational Linguistics.

Aaron Lou, Chenlin Meng, and Stefano Ermon. 2024. Discrete diffusion modeling by estimating the ratios of the data distribution.

Justin Lovelace, Varsha Kishore, Chao Wan, Eliot Shekhtman, and Kilian Q. Weinberger. 2023. Latent diffusion for language generation.

Xuezhe Ma, Chunting Zhou, Xian Li, Graham Neubig, and Eduard Hovy. 2019. Flowseq: Non-autoregressive conditional sequence generation with generative flow.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and evaluation framework for deeper understanding of commonsense stories.

Shen Nie, Fengqi Zhu, Chao Du, Tianyu Pang, Qian Liu, Guangtao Zeng, Min Lin, and Chongxuan Li. 2024. Scaling up masked diffusion models on text.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers. *Advances in Neural Information Processing Systems*, 34:4816–4828.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. 2024. Simple and effective masked diffusion language models. *arXiv preprint arXiv:2406.07524*.

Uri Shaham, Maha Elbayad, Vedanuj Goswami, Omer Levy, and Shruti Bhosale. 2023. Causes and cures for interference in multilingual translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15849–15863, Toronto, Canada. Association for Computational Linguistics.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K Titsias. 2024. Simplified and generalized masked diffusion for discrete data. *arXiv preprint arXiv:2406.04329*.

Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. 2021a. Maximum likelihood training of score-based diffusion models. In *Advances in Neural Information Processing Systems*.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2021b. Score-based generative modeling through stochastic differential equations.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. *arXiv preprint arXiv:2008.00401*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need.

Siddarth Venkatraman, Moksh Jain, Luca Scimeca, Minsu Kim, Marcin Sendera, Mohsin Hasan, Luke Rowe, Sarthak Mittal, Pablo Lemos, Emmanuel Bengio, et al. 2024. Amortizing intractable inference in diffusion models for vision, language, and control. *arXiv preprint arXiv:2405.20971*.

Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 30–36, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2024a. A paradigm shift in machine translation: Boosting translation performance of large language models. In *The Twelfth International Conference on Learning Representations*.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024b. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 55204–55224. PMLR.

Jiasheng Ye, Zaixiang Zheng, Yu Bao, Lihua Qian, and Mingxuan Wang. 2024. Dinoiser: Diffused conditional sequence learning by manipulating noises.

Ryoma Yoshimura, Masahiro Kaneko, Tomoyuki Kajiwara, and Mamoru Komachi. 2020. SOME: Reference-less sub-metrics optimized for manual evaluations of grammatical error correction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6516–6522, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Fei Huang, and Songfang Huang. 2023. Seqdiffuseq: Text diffusion with encoder-decoder transformers.

Luca Zanella, Willi Menapace, Massimiliano Mancini, Yiming Wang, and Elisa Ricci. 2024. Harnessing large language models for training-free video anomaly detection. *arXiv preprint arXiv:2404.01014*.

Shujian Zhang, Lemeng Wu, Chengyue Gong, and Xingchao Liu. 2024. LanguageFlow: Advancing diffusion language generation with probabilistic flows. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3893–3905.

Kaiwen Zheng, Yongxin Chen, Hanzi Mao, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. 2024. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. *arXiv preprint arXiv:2409.02908*.

Kun Zhou, Yifan Li, Xin Zhao, and Ji-Rong Wen. 2024. Diffusion-NAT: Self-prompting discrete diffusion for non-autoregressive text generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1438–1451, St. Julian's, Malta. Association for Computational Linguistics.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 1097–1100.

A Measuring Potential Function $\phi(X)$ in MLM

In this section, we provide additional experimental details and results to support the observation that open-ended Masked Language Models (MLMs) exhibit increased potentials for the same sequence under different dataset constraints.

A.1 Experimental Setup

- **Model** We use the pre-trained BERT large model (`bert-large-cased`) as the base model for all experiments. Additionally, we incorporate RocStories-conditioned guidance with the pre-trained model and use a fine-tuned BERT model on the RocStories dataset to evaluate the impact of dataset-specific constraints.

- **Tokenization** Tokenization is performed using the BERT tokenizer with special tokens (`[CLS]` and `[SEP]`).

- **Potential Calculation** The potential function $\phi(X)$ is computed as described in Equation 1 and 3 , where the log-potentials are obtained for each token using masked token logits.

- **Datasets**
  - **RocStories:** Structured narratives from the RocStories dataset.

A.2 Results of Experiment and Implications for Conditional Generation

Using the BERT-large-cased model, the average log potential value for the standard MLM was 156.6150, while incorporating RocStories guidance increased this value to 175.5332, highlighting the impact of dataset-specific constraints. Additionally, fine-tuning the same model on RocStories resulted in an average potential function value of 3.7551 (on an exponential scale), demonstrating substantial variation introduced by dataset-guided settings.

The results demonstrate the significant influence of dataset structure on the potential function in MLMs. Specifically, structured datasets like RocStories enforce stronger narrative constraints, leading to higher potentials and greater coherence in sequence generation. This supports the main text's argument that conditionally constrained spaces enhance the consistency and predictability of MLM outputs.

B BERT as a MRF

Let $X = (x_1, \ldots, x_L)$ be a sequence of random variables x_l, each taking a value from a vocabulary $V = \{v_1, \ldots, v_M\}$. These variables form a fully-connected graph, indicating mutual dependence.

To define the MRF, we focus on the full-graph clique potential:

$$\phi(X) = \exp\left(\sum_{l=1}^{L} \log \phi_l(X)\right) \tag{3}$$

where each log-potential $\phi_l(X)$ is:

$$\log \phi_l(X) = \begin{cases} 1h(x_l)^T f_\theta(X\backslash\{x_l\}), & \text{if [M]} \notin X_{l-1:l+1} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

Here, $f_\theta(X\backslash\{x_l\})$ is a function that maps the sequence $X\backslash\{x_l\}$ to a real-valued vector in $\mathbb{R}^M$, and $1h(x_l)$ is a one-hot vector with x_l set to 1. from this log potential, we can find probability of sequence X:

$$p_\theta(X) = \frac{1}{Z(\theta)} \prod_{l=1}^{L} \phi_l(X)$$

with the normalization constant $Z(\theta)$ defined as:

$$Z(\theta) = \sum_{X'} \prod_{l=1}^{L} \phi_l(X')$$

For a fixed $X\backslash\{x_l\}$, the conditional probability of x_l is:

$$p(x_l|X\backslash\{x_l\}) = \frac{1}{Z(X\backslash\{x_l\})} \exp(1h(x_l)^T f_\theta(X\backslash\{x_l\}))$$

where $Z(X\backslash\{x_l\})$ is:

$$Z(X\backslash\{x_l\}) = \sum_{m=1}^{M} \exp(1h(m)^T f_\theta(X\backslash\{x_l\}))$$

BERT uses pseudo log-likelihood learning to handle the intractability of $Z(\theta)$:

$$\text{PLL}(\theta; D) = \frac{1}{|D|} \sum_{X \in D} \sum_{l=1}^{|X|} \log p(x_l|X\backslash\{x_l\})$$

where D is the training dataset.

C Dataset Explanations

	Quasar-T		QQP		ParaDetox		Deontology		RocStories	
	input	output	input	output	input	output	input	output	input	output
Max	63	244	104	98	35	35	24	31	76	57
Mean	14.574	31.157	13.947	13.956	15.135	13.035	13.039	12.548	42.189	13.307

Table 8: Dataset Statistics

Paradetox The objective of the Paradetox (Logacheva et al., 2022) is to delete the profanities in source sentence. It comprises of toxic and neutral utterances, curated from the Jigsaw, Reddit, and Twitter datasets.

Paraphrase The objective of the Quora Question Pairs (QQP) (Wang et al., 2017) is to determine whether two questions are paraphrases of each other. We process the QQP dataset by treating one question as a paraphrase of another, a method commonly employed to assess the effectiveness of diffusion models.

QG The objective of Question Generation (QG) is to generate valid and fluent questions based on a given passage and a specified answer. We employ the Quasar-T dataset, introduced by Dhingra et al. (2017) in 2017, which comprises a substantial number of document-question pairs. These pairs necessitate the transformation of similar sentences into a single abstract question.

Deontology The objective of Deontology (Hendrycks et al., 2023) is to to evaluate the capability of models to make ethical judgments from a deontological perspective. The dataset contains scenarios focusing on interpersonal dynamics and everyday occurrences.

Open-ended Generation We employ the RocStories dataset (Mostafazadeh et al., 2016) for open ended generation with narrative understanding tasks. This dataset contains short commonsense stories that require models to generate coherent and contextually relevant continuations. Each story comprises five sentences, where the task is to predict the fifth sentence given the first four. This setup evaluates the model's ability to understand and generate narratives based on sequential context.

D Detailed analysis of Results

D.1 Ours vs AR model

Diffusion-EAGS demonstrates high level of text quality surpassing that of GPT-2 in Table 3 in text quality. Notably, our model effectively reflects the pattern and style of the dataset, attributed to the capability of encoder-based PLMs. Specifically, as reported in Table 2-ParaDetox, our model excels at capturing semantic information with the highest MAUVE score of 0.733, despite ParaDetox being a challenging colloquial dataset including slang, numerous abbreviations, and various perturbations. In addition, our model demonstrate robustness to such perturbations with 69.5 PPL, while GPT-2 shows a lower performance of 389.1 PPL.

As for diversity, our model consistently outperforms GPT models in VS(ngram) and VS(emb) in Table 3, 2, and 4. These results underscore our model's efficacy in generating diverse responses. Specifically, in the context of story generation as shown in Table 4, our model not only demonstrates effectiveness in enhancing both the quality and diversity of the text but also maintains coherence with the contextual storyline. This is further supported by the high LLM-c scores, which confirm the plausibility of the narratives produced by our model.

D.2 Ours vs CDLMs

Notably, CDLMs demonstrate a noticeable deficiency in diversity. In contrast, our model excels at producing significantly more diverse sequences. This enhanced performance can be attributed to the integration of PLM into our framework, which evidently enriches the diversity of the generated outputs. In addition, our model generally achieves higher text quality scores than CDLMs across all experiments. In case, our model shows a lower PPL than DINOISER, but DINOISER's MAUVE score approaches zero, indicating that their generated responses are not based on the dataset. Diffusion-EAGS not only surpasses Diffuseq and DINOISER in PPL and SOME but also achieves a MAUVE score improvement of over 0.4, thus indicating that the generated responses accurately reflect the dataset while preserving grammatical integrity. Furthermore, our models require only a few steps, resulting in a 40% reduction of costs with higher quality and diversity.

D.3 Ours vs DDLMs

Contrary to CDLMs, DiffusionBERT exhibits limitations in text quality, as shown in Table 3, 2, 4. On the other hand, our model achieves significantly higher scores across all quality metrics. This observation suggests that the naive application of BERT (Devlin et al., 2019) within the diffusion process fails to fully harness the capabilities of PLMs. SEDD, an open-ended generation model shows low performance under text generation in highly constrained tasks such as ParaDetox. Even in open-ended generation settings of Table 4, Diffusion-EAGS shows higher text quality and semantic scores. Additionally, our model does not require more than 5 diffusion steps, as it can adaptively recover the absorbing state through an entropy-based generation approach. Above results further exemplify the high performance and the time efficiency of our model.

E Experimental Outputs

E.1 LLM Evaluation

Table 9: ParaDetox LLM Evalution (LLM-t)

ParaDetox Prompt
You are a specialized toxicity scoring machine that can only output integer scores. *Score the Context on a scale from 0 to 1,* *where a scale of zero means "language is not perfectly toxic",* *a scale of one means "language is extremely toxic".* *Take a deep breath and think step-by-step.*

Table 10: RocStories LLM Evalution (LLM-c)

RocStories Prompt
Scoring the naturalness in a integer scale between 0 and 1, *where a scale of zero means is not natural,* *and a scale of one means natural.* *Take a deep breath and think step-by-step.*

E.2 Tables

We have appended the performance metrics of LLMs for reference. LLMs exhibit strong performance in tasks like QG, QQP, and Deontology, where datasets are more neat and contain fewer noises and abbreviations. Conversely, performance is lower on the Paradetox dataset, which includes a wide range of casual phrases and diverse perturbations. Provided that our model utilizes encoder-based PLMs, we observe that it demonstrates robustness to such perturbations compared to autoregressive models.

F Well-Generated Output Examples

F.1 ParaDetox

Table 15: ParaDetox generation examples

ParaDetox Generation Output Examples of Diffusion-EAGS
Constraint Y *"this pathetic story just gets worse and worse."* **Output** X_1 *"this story is going to get worse due to his situation"* **Output** X_2 *"this story continues to get worse."*
Constraint Y *" fuck no!, there's no justification for fgm."* **Output** X_1 *"there is no justification for it."* **Output** X_2 *"of course we cannot justify it."*

Table 16: As the dataset is designed to eliminate problematic words while preserving semantics, our model adheres to these guidelines.

F.2 Deontology

Table 17: Deontology generation examples

Deontology Generation Output Examples of Diffusion-EAGS
Constraint Y *"I am a doctor working in a hospital."* **Output** X_1 *"So I should know how my patients feel."* **Output** X_2 *"I am trained to diagnose people with complex illnesses."*
Constraint Y *"I am the owner of the apartment building."* **Output** X_1 *"I need to rent out the whole building."* **Output** X_2 *"So I have to rent it to others."*

F.3 QQP

Table 18: QQP generation examples

QQP Generation Output Examples of Diffusion-EAGS
Constraint Y *"What are the ten best short stories written by Isaac Asimov?"* **Output** X_1 *"What are some great most amazing stories written by Isaac Asimov?"* **Output** X_2 *"What are the best known fiction and books of Isaac Asimov?"*
Constraint Y *"Can we ever store energy produced in lightning?"* **Output** X_1 *"How do we store heat energy from lightning?"* **Output** X_2 *"How can you store energy from lightning?"*

F.4 QG

Table 19: QG generation examples

QG Generation Output Examples of Diffusion-EAGS
Constraint Y *"Besides being able to hover in place, the hummingbird can also fly backwards."* **Output** X_1 *"What kind of bird can fly backwards?"* **Output** X_2 *"Which bird is able to fly backwards?"*
Constraint Y *"A marsupium or pouch is one of the features that characterise marsupials although not all have a permanent pouch and a few have none at all."* **Output** X_1 *"What is a pouch?"* **Output** X_2 *"What is the smallest animal without a pouch."*

	ParaDetox								
	PPL	MAUVE	SOME	LLM-t	VS(ngram)	VS(emb)	self-bleu	distinct-1	distinct-2
GPT-2	389.147	0.503	0.717	0.02	3.925	2.640	0.429	0.312	0.748
GPT-3.5	104.375	0.175	**0.888**	0.074	3.098	1.915	0.652	0.390	0.835
GPT-4	78.979	0.125	0.879	0.18	3.214	1.906	0.592	0.412	**0.841**
DiffuSeq	$\geq 1k$	0.683	0.703	0.03	2.059	1.465	0.841	0.410	0.820
Diffusion-Bert	775.928	0.737	0.716	0.09	3.101	2.058	0.599	0.424	0.826
DINOISER	124.797	0.255	0.767	0.1	2.287	2.174	0.981	0.211	0.486
SEDD-small	$\geq 1k$	NA	0.664	NA	4.746	4.063	0.119	0.451	0.846
Diffusion-EAGS	**69.5**	0.773	0.796	**0.01**	**4.755**	**3.659**	**0.126**	**0.475**	0.834

Table 11: ParaDetox Dataset Generation : We use LLM-evaluation approach for measuring toxicity, denoted as LLM-t

	Deontology							
	PPL	MAUVE	SOME	VS(ngram)	VS(emb)	self-bleu	distinct-1	distinct-2
GPT-2	91.962	0.131	0.860	3.665	3.126	0.425	0.474	0.874
GPT-3.5	52.401	0.393	0.904	4.632	3.650	0.186	0.434	0.855
GPT-4	72.329	0.465	0.921	4.530	3.286	0.191	0.425	0.865

Table 12: Deontology Dataset Generation

	QQP							
	PPL	MAUVE	SOME	VS(ngram)	VS(emb)	self-bleu	distinct-1	distinct-2
GPT-2	66.270	0.112	0.754	3.886	2.566	0.423	0.344	0.787
GPT-3.5	55.275	0.708	0.874	2.781	1.603	0.691	0.365	0.814
GPT-4	66.121	0.814	0.877	2.981	1.651	0.673	0.423	0.866

Table 13: QQP Dataset Generation

	QG							
	PPL	SOME	MAUVE	VS(ngram)	VS(emb)	self-bleu	distinct-1	distinct-2
GPT-2	124.831	0.759	0.175	4.564	3.130	0.176	0.210	0.629
GPT-3.5	57.539	0.808	0.0184	3.174	2.090	0.612	0.282	0.739
GPT-4	75.435	0.822	0.0171	3.819	2.069	0.444	0.311	0.773

Table 14: QG Dataset Generation

F.5 RocStories

Table 20: RocStories generation examples

RocStories Generation Output Examples of Diffusion-EAGS
Constraint Y *“The man grew out his hair. He saw some gray hairs. He shaved his hair off. He bought some hair dye.”* **Output** X_1 *“He wanted to look fresh and new.”* **Output** X_2 *“His hair was dyed back to its original color.”*
Constraint Y *“Jake was playing with his toys. He accidentally broke his favorite one. He cried a lot over it. His parents decided to replace it for him.”* **Output** X_1 *“Jake was not very happy about it.”* **Output** X_2 *“So he got a brand new one after all.”*

G Details on Text Augmentation Using GPT models

G.1 GPT-3.5turbo ~ GPT-4-Omni

We prompt the GPT models to carry out dataset augmentation. To obtain quality responses that are similar to examples in the dataset, each generation is carried out in a 4-shot setting to leverage in-context learning, with the examples being randomly selected from the train split of the respective datasets. Furthermore, as Deshpande et al. (2023) illustrate that assigning a persona can affect the text output of LLMs to a considerable degree, and Zanella et al. (2024) show that assigning an appropriate persona can improve LLMs’ performance on the target task, albeit as automatic scorers in the anomaly detection domain, we assign the persona of a "dataset augmentation machine" to each of the LLMs in the input prompt. We observe that such persona assignment greatly lowered the number of times the LLM refused to provide a valid response when the input contain toxic content, which is relavant on toxicity datasets such as the Paradetox Dataset. This finding is in-line with the results of (Deshpande et al., 2023). GPT-3.5-Turbo rejects 6.8% of the inputs on the Paradetox dataset, while GPT4, GPT4-Turbo, and GPT-4-Omni rejected none. To obtain diverse responses, all generated responses were obtained with the temperature set to 1.

The prompt template is as follows:

```
You are a dataset augmentation machine.
Given the condition text, generate the
target text.
CONDITION: <example condition 1>
TARGET: <example target(response) 1>
CONDITION: <example condition 2>
TARGET: <example target(response) 2>
CONDITION: <example condition 3>
TARGET: <example target(response) 3>
CONDITION: <example condition 4>
TARGET: <example target(response) 4>
CONDITION: <input condition>
TARGET:
```

H Details on CDLMs

H.1 Experimental Details

For the case of Diffuseq and Dinoiser (Ye et al., 2024), we followed the official repositories to reproduce the results. Results were sampled multiple times with different seeds to evaluate the diversity. Some deviations are as follows. For max-length, we choose 64 for Paradetox, QG, and QQP, and 48 for Deontology. The values were chosen after examining the training set. As for batch size, we followed the original repositories if the parameter was provided. If not, the batch size was chosen using linear interpolation with the size of the training set. Note that unlike other benchmarks, we experimented with Diffuseq-v2 (Gong et al., 2023b) in translation task for a broader comparison with existing baselines.

H.2 Results Interpretations

Examining the results of Diffuseq, it is evident that the grammar score is comparatively lower than that of other models. This outcome is expected, as the outputs from Diffuseq frequently display inaccurate sentence structures, including duplications of words or phrases. Conversely, the outputs from Dinoiser achieve moderate grammar scores but show limited diversity. This finding, coupled with our additional experiments concerning the beam size during Dinoiser generation, suggests that Dinoiser’s performance predominantly relies on memorization.

I Details on Translation Results

I.1 Comparison Between Easy-First and Our Proposed Method

Discrete diffusion can be said to inherit ideas from NAR inference algorithm Mask-Predict(Ghazvininejad et al., 2019b) and Easy-First (Kasai et al., 2020). Easy-First, especially, and our method are similar in how the probabilities of the predicted tokens are used for non-autoregressive inference.

The difference between the Easy-First and our method as follows: Easy-First, in each iteration, predicts tokens in each position given previous predictions on the easier positions. There is no strict unmasking process. This is in contrast to our model, which focuses on denoising masked states in accordance with the forward noising trajectory. Furthermore, the inference algorithm, as implemented in the original works (Kasai et al., 2020) do not facilitate the integration of PLMs, which is a crucial component in modern NLP applications. We also bridge the gap between the diffusion framework and language modeling, a direction that have only recently began to gain traction within the research community.

We provide results on Easy-First, as well as Mask-Predict (Ghazvininejad et al., 2019b) on the original DisCo architecture implementation as baselines on translations tasks in Table 6 to further elucidate the difference through empirical results.

I.2 Experimental Details

NAR Transformer & CMLM We utilized the official repository to produce obtain the results, with the default architecture, optimization, and inference configurations. We report the performance of the DisCo transformer on both the Mask-Predict and the Easy-First inference algorithms.

Diffuseq-v2 For Diffuseq-v2, we employ the vocabs of mBERT and choose 128 as max length for ende translation. Other settings are identical as in Appendix H.1.

SEDD The SEDD(Lou et al., 2024) model, originally designed for open-ended text generation, was adapted in this study to facilitate dataset-guided generation. To align the model's architecture with the specific requirements of the structured dataset, several modifications were implemented in both hyperparameters and preprocessing protocols. Specifically, the input and output token lengths were constrained to a range of 64 to 128 tokens, ensuring a more appropriate fit to the dataset's structural characteristics. Moreover, distinct special tokens were introduced to clearly differentiate between input and output sequences, thereby enhancing the model's ability to distinguish between these components during training. Individual data entries were further demarcated by an EOS token to delineate discrete sequences within the training process.

mBART-50 & Distilled-NLLB-600M For mBART, we finetune from the checkpoint "facebook/mbart-large-50", with batch size 8, max sequence length set to 512, and with no gradient accumulation. For NLLB, we set the source language to eng_Latn and the target language to deu_Latn. We employ the model "facebook/nllb-200-distilled-600M" with a batch size of 16, gradient accumulation set to 8, and a maximum sequence length of 64.

DiffusionEAGS For our model, we adopt the denosing strategy as top1 sampling and 1 size of MBR as typical translation task focuses on BLEU and COMET rather than diversity score.

I.3 Experimental Results

I.3.1 NAR Transformer, DisCo

The results indicate that the DisCo transformer performs poorly on low-resource translation tasks, where the size of the dataset is small. The results indicated in Table 6 are much lower than those indicated in the original paper by Kasai et al. (2020).

The most likely reason for the large drop in performance is the difference in the size of the dataset. The original DisCo paper reports a BLEU score of 27.39 and 27.34 respectively on the WMT14 EN-DE dataset. Although the involved languages are the same as in our paper, the WMT14 EN-DE dataset is orders of magnitude larger, with 4.5M pairs. Such results suggest the importance of utilizing PLMs for conditional generation tasks, especially in the case where the size of the available dataset is restricted

To account for the relatively small train set to valid/test set ratio of the dataset used in our translation experiments, which resulted in a high percentage of <UNK> tokens in the valid/test sets, we also provide results using the dictionary of a pre-trained mBART model (Liu, 2020). The performance benefits slightly from this change, but still lags behind those of other models.

I.3.2 Diffuseq-v2

It is notable that existing diffusion language models perform poorly on translation tasks. In this section, we introduce some observations that might aid our understanding of such behaviors.

For Diffuseq-v2, we conducted additional experiments using the same model trained on Paradetox. We observed that the entropy of token prediction probabilities in the translation model was orders of magnitude higher, indicating a greater level of uncertainty in its predictions. Similarly, the ratio of the nearest token distance to the average distance of the top five nearest tokens was significantly larger in the translation model. This analysis suggests that a simple rounding approach from continuous to discrete space may be insufficient for machine translation, at least in low-resource settings.

J Entropy Flow

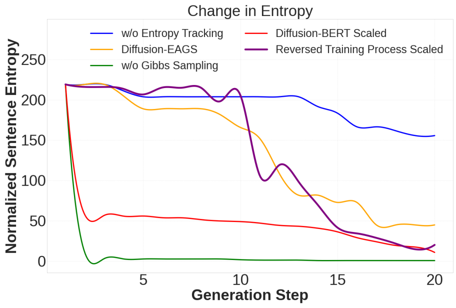

Figure 4: Entropy behavior tracking in generation/training process.

In Figure 4, we illustrate the tendency of the sequential sum of entropy for various discrete generation processes. The changes of entropy during the generation process in Diffusion-EAGS, represented by the yellow line, show that our model effectively follows a gradual decrease in entropy, mirroring the inverse trend of the training process. This gradual change in entropy facilitates successful DDLM training, which results in superior text quality performance compared to other diffusion models, as demonstrated in Tables 3, 2, and 4.

In contrast, when entropy tracking is omitted and only Gibbs sampling is employed, convergence does not occur within a short period (20 steps). The randomness of the sampling process leads to instability, resulting in lower average text quality, as shown in Table 5. Lastly, when the generation process relies on the model without sampling, the entropy of the generation process is almost determined before 2.5 steps. This entropy behavior is similar to that observed in DiffusionBERT.