Enforcing Paraphrase Generation via Controllable Latent Diffusion

Anonymous ACL submission

Abstract

Paraphrase generation aims to produce high-quality and diverse utterances of a given text. Though state-of-the-art generation via the diffusion model reconciles generation quality and diversity, textual diffusion suffers from a truncation issue that hinders efficiency and quality control. In this work, we propose *L*atent *D*iffusion *P*araphraser (LDP), a novel paraphrase generation by modeling a controllable diffusion process given a learned latent space. LDP achieves superior generation efficiency compared to its diffusion counterparts. It facilitates only input segments to enforce paraphrase semantics, which further improves the results without external features. Experiments show that LDP achieves improved and diverse paraphrase generation compared to baselines. Further analysis shows that our method is also helpful to other similar text generations and domain adaptations. Our code and data are available at https://anonymous.4open.science/r/8F72 .

1 Introduction

Paraphrase generation aims to produce semantically equivalent sentences in varied linguistic forms. It's versatile in generation tasks such as text summarization (Zhou and Bhat, 2021; Dong et al., 2017) and question answering (Buck et al., 2017). Techniques such as data augmentations (Kumar et al., 2019) also involve paraphrasing.

Though mainstream paradigms have achieved success, they still struggle to balance generation quality and diversity. The deterministic paradigm via the encoder-decoder model (Vaswani et al., 2017) focuses on high-quality generation instead of diversity. Such paradigm is further improved in diversity by enforcing explicit external features of high-level semantics shared by paraphrases, such as syntax (Sun et al., 2021; Bao et al., 2019) and examplars (Yang et al., 2021; Chen et al., 2019). However, external features for diversity are not always available. On the other hand, the variational paradigm promotes diversity via variational autoencoders, which model the latent distribution shared by diverse contextual representations (Bao et al., 2019; Du et al., 2022). However, its sampling nature given limited Gaussian distributions hinders the generation quality of complex language patterns despite additional keyword guidance (Chen et al., 2022).

Recently, a novel generation paradigm via the diffusion probabilistic model (Ho et al., 2020, DPM) achieves state-of-the-art generation in both quality and diversity for images and speech. The DPM is the generation that morphs toward high-quality data through numerous rounds of continuous Markov transitions. Though DPM shares similar stochasticity with the variational generation, it does not fall short in quality. Additionally, its neat interventions enable the continuous diffusion transitions to meet the quality required by versatile data generations (Nichol and Dhariwal, 2021), which plays a vital role in diffusion implementation. Therefore, we consider the diffusion paradigm to fit paraphrase generation, where its controllability also meets the intuition of possible intervention in traditional paradigms.

Lately, Diffusion-LM (Li et al., 2022) and D3PM (Austin et al., 2021) further cater to text generation via an additional discrete sampling called 'rounding' process. The 'rounding' essentially bridges the continuous diffusion representations with corresponding discrete tokens, as arbitrary diffusion intervals are truncated to embeddings of valid texts with additional decodings. However, 'rounding' introduces decoding overhead, thus hindering high-efficiency generation by SOTA diffusion implementations (Ye et al., 2023; Karras et al., 2022). Furthermore, 'rounding' also introduces truncation errors when arbitrary diffusion intervals are rounded to specific token embeddings, further hindering possible intervention. Therefore, we consider circumventing the truncation issue to

improve efficiency and enable generation interventions when paraphrasing via text diffusion.

In this work, we propose *L*atent *D*iffusion *P*araphraser (LDP), which can enforce semantics by only input segments instead of external features. LDP adopts the latent space from a given encoder-decoder framework, which offers more efficacy than raw features for diffusion as suggested by Rombach et al. (2022); Lovelace et al. (2022). The off-the-shelf encoder and decoder bridge the continuous diffusion process with corresponding discrete texts, thus LDP prevents the intermediate roundings required by diffusion on raw text, which offers generation efficiency. Furthermore, removing roundings enables state-of-the-art control for diffusion steps, where we further utilize only input segments rather than external features to enforce semantics for improvement. Experiments show that LDP achieves better and faster paraphrase generation than its diffusion counterparts on various datasets. Further analysis shows that our methods are helpful to other similar text generations and domain adaptation.

Our contributions can be summarized as follows:

- We propose a novel paraphrase generation called LDP, which improves generation quality and diversity. LDP circumvents 'rounding' thus more efficient compared to its diffusion counterparts.
- LDP can enforce paraphrase semantics with only input segments instead of external features, which further improves results.
- Analysis shows that our method is also helpful in other similar text scenarios such as question generation and domain adaptation.

2 Preliminary

2.1 Paraphrase Generation

Paraphrase refers to the diverse utterances that keep the original semantic. Paraphrase generation is crucial in several downstream natural language processing (NLP) tasks. Though methods based on deterministic seq2seq framework have achieved success (Vaswani et al., 2017; Sancheti et al., 2022; Yang et al., 2019), the nature of maximum likelihood estimation hinders the generation diversity. Some researchers promote generation diversity by enforcing explicit external features of high-level semantics shared by paraphrases such as syntactic structures or exemplar syntax (Hosking et al., 2022; Yang et al., 2022), which are not easily accessible. Others turn to variational generation to fit the shared latent distribution of diverse textual representations (Bowman et al., 2015; Du et al., 2022). However, variational generations sacrifice the quality for its diversity.

2.2 Latent Diffusion Models with Control

The diffusion probabilistic model (Ho et al., 2020, DPM) is a Markov chain of variational reconstruction of the original inputs z_0 given data distribution $q(z)$ from Gaussian-distributed noise. Specifically, the DPM is trained by sampling from a Markov noising process $P(z_{t+1}|z_t) \sim \mathcal{N}(z_{t+1}; \sqrt{1-\beta_t}z_t, \beta_t\mathcal{I})$ scheduled by series of noise scales $\beta_t \in (0,1)$, where the β_t is considered the standard deviation of the step-wise transition distribution $\sqrt{\beta_t} = \sigma_t$ at step $t \in [0,T]$. Rombach et al. further proposes the latent diffusion where the diffusion process is introduced to the latent space of a well-trained encoder-decoder framework. The latent diffusion achieves better results for video (He et al., 2023), audio (Liu et al., 2023), and image synthesis (Lai et al., 2023) since diffusion upon encoded latent features proves more efficacy than that upon raw representations.

Training a diffusion model follows the intuition to fit reconstruction from noised z_t to its original z_0 along a T-step Markov-Gaussian noising process:

$$\mathcal{L} = \mathbb{E}_{t\sim[0,T]}\left[\|z_\theta(z_t,t) - z_0\|_2^2\right], \quad (1)$$

where $z_\theta(\cdot)$ is a reconstruction neural net given noised z_t and noising step t. Ho et al. (2020) further deducted t-steps Markov-Gaussian noising process into a one-step noising, leading to a closed form z_t by given noising step t:

$$z_t = \sqrt{\overline{\alpha}_t}z_0 + \sqrt{1-\overline{\alpha}_t}\epsilon, \epsilon \sim \mathcal{N}(0,\mathcal{I}), \quad (2)$$

where $\overline{\alpha}_t = \Pi_{i=1}^{t}(1-\beta_i)$, as t determines the noise schedule β_t.

The corresponding diffusion generation iteratively morph from pure Gaussian noise to valid data by the learned reconstructor $z_\theta(\cdot)$. That is, given a fully optimized $z_\theta(\cdot)$, the generation starts from pure Gaussian sample $z_T \in \mathbb{R}^{l\times d} \sim \mathcal{N}(0,I)$ with a roughly reconstructed $\hat{z}_0$, then iteratively noise and denoise by diminishing noise scales determined by β_t. Such iteration can be conducted via various diffusion sampling algorithms such as ancestral sampling (Ho et al., 2020), DDIM sampler (Song et al., 2020a), or ODE solvers(Lu et al., 2022a,b).

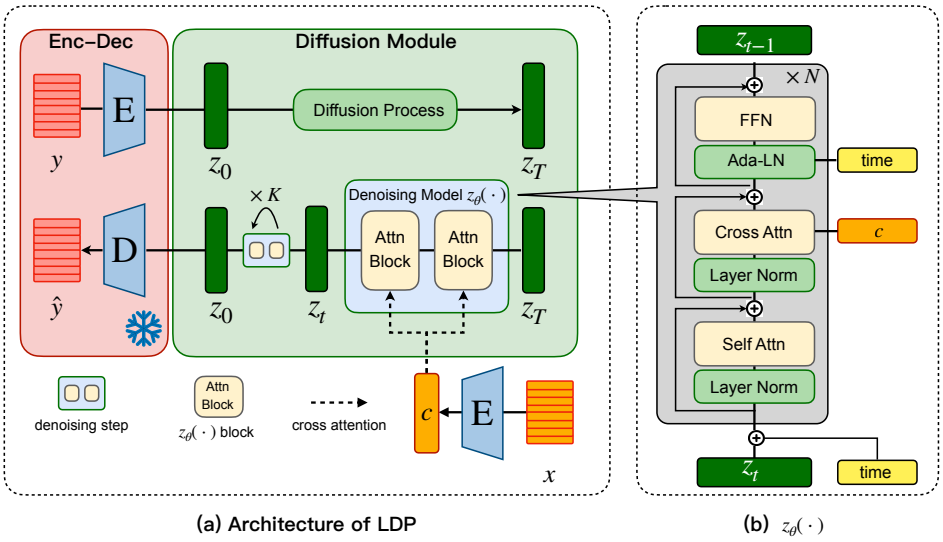

Figure 1: Overview of LDP. (a) Model architecture. LDP consists of an encoder-decoder framework (pink) and a diffusion module (green). The encoder (E) and decoder (D) are frozen to bridge the continuous diffusion process with corresponding discrete texts. (b) Detailed architecture of denoising model $z_\theta(\cdot)$

The iterative diffusion generation can be intervened by plug-and-play modules with minor overheads, where the state-of-the-art modeling is ControlNet (Zhang et al., 2023). ControlNet finetunes a trainable copy of the DPM $z'_\theta(\cdot)$ to cater to versatile generations while freezing the learned DPM $z_\theta(\cdot)$ to maintain harmless adaptation. The trainable copy $z'_\theta(\cdot)$ takes in additional control signals to the DPM via zero-convolution layers, i.e., convolution layers initialized by zero weight and bias. The controller inputs δ are fused with original diffusion steps by Eq 3:

$$z_{t+1} = z_\theta(z_t, t) + \text{zero\_conv}(z'_\theta(z_t + \text{zero\_conv}(\delta), t)), \quad (3)$$

where δ enables vague yet versatile guidance such as Canny edges and poses (Zhang et al., 2023).

3 Methodology

In this section, we detail *L*atent *D*iffusion *P*araphraser (LDP). LDP models the diffusion process upon the latent space of a pretrained encoder-decoder framework, where mere input segments can further enforce paraphrase semantics for improvement.

3.1 Latent Diffusion Paraphraser

The overall pipeline is shown in Fig 1(a). LDP consists of an encoder-decoder framework to provide a learned latent space and a diffusion module. The encoder and decoder with a learned latent space bridge the continuous diffusion process with corresponding discrete texts once and for all at the beginning and the end of the generation, instead of step-wise rounding. Notably, LDP is compatible with the mainstream encoder-decoder framework as long as it bijectively maps the texts with the corresponding latent representations.

To make life easier, we adopt BART (Lewis et al., 2019) for illustration, which is an off-the-shelf pretrained language model that encodes and decodes arbitrary text with its corresponding latent representation. Given a text sequence $x = \{x_1, x_2, \ldots, x_l\}$, the BART encoder E encodes it into a d-dimentional latent representation $z = \text{E}(x), z \in \mathbb{R}^{l \times d}$ for diffusion process, while the decoder D yields corresponding text sequence given the latent representation $y \approx \text{D}(z) = \text{D}(\text{E}(x))$. The model utilizes T5 relative positional embeddings (Raffel et al., 2020). Input features for diffusion are normalized by training data features, where we adopt BART encodings for mean and standard deviation (Rombach et al., 2022). Concordantly, the normalization is reversed before text decoding. The encoder and decoder are frozen during diffusion training, thus leaving the reconstruction network $z_\theta(\cdot)$ the only trainable parameters.

Given paraphrase pair $\langle x, y \rangle$, we then train the end-to-end diffusion models for encoder latent space, where the reconstruction network is parameterized by $z_\theta(z_t, c, t)$ with addition source encoding $c = E(x)$ as input. $z_\theta(\cdot)$ consists of N layers

of pre-layer norm transformer blocks. As shown in Fig 1(b), each layer consists of a self-attention encoding for z_t, followed by a cross-attention access of the encoded source c and a time step interpolation via a feedforward layer. The time step t is embedded as a d-dimensional vector, then its interpolation is preprocessed by AdaLN (Xu et al., 2019; Peebles and Xie, 2022, adaptive layer norm) instead of generic layer norm. AdaLN regresses the layer-wise normalization scale and shifts from the sum of time embedding and encoded input features. The network layers are activated by GeGLU (Shazeer, 2020) following SOTA transformer implementation (Raffel et al., 2020).

$z_\theta(\cdot)$ is optimized by reconstruction loss in Eq 1. We uniformly sample time step t for arbitrary noised representation z_t by Eq 2, where t determines the noise scale β_t by noise schedule (Ho et al., 2020; Nichol and Dhariwal, 2021). We also apply sentence-level condition dropout during training to ensure the model's unconditional language generation, that is, to replace the source sentence representation with trainable null tokens $y_\varnothing$ with probability $p = 0.1$.

The optimized $z_\theta(\cdot)$ is implemented in SOTA diffusion samplers (Song et al., 2020b; Lu et al., 2022a,b), which will morph pure Gaussian noise z_T into the latent representation $\hat{z}_0$ of the given source text. D further decodes $\hat{z}_0$ for text output. Unlike generation with length prediction, LDP determines the sequence length by end-of-sequence label automatically.

3.2 Semantic Enforcing by Controller

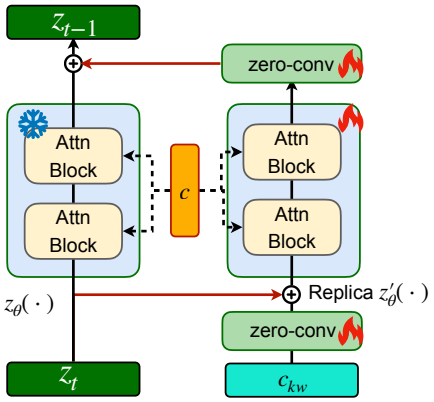

Figure 2: Architecture of LDP controller, where c_{kw} indicates the encoded keyword segments. We freeze the learned $z_\theta(\cdot)$, then finetune the replica $z'_\theta(\cdot)$

Diffusion generation can introduce additional controls via ControlNet (Zhang et al., 2023), which is a step-wise plug-and-play controlling framework. It has proved its efficacy by manipulating image generation given vague constraints such as canny edges or skeleton poses. By removing step-wise rounding, LDP circumvents the truncation issue against control signals, where we consider paraphrase generation can harness input segments as the vague constraints likewise.

Intuitively, the paraphrase can be enforced by mere segments of vital semantics via a controller for better generation. Therefore, we leverage input tokens above a certain length as keywords, enforcing the diffusion generation on given semantics to improve paraphrase. Our implementation is shown in Fig 2. We sample from the longest 15% tokens in a given sentence as keywords and mask the remaining parts with placeholder <M> as semantic segments. The semantic segments are then encoded by the same encoder used by the original DPM for controller inputs. The controller block $z'_\theta(\cdot)$ is a trainable copy of the well-trained $z_\theta(\cdot)$, while $z_\theta(\cdot)$ is frozen. The controller inputs are fused with original diffusion generation by Eq 3 as:

$$\begin{aligned}\hat{z}_0 =& z_\theta(z_t, c, t) + \\ & \text{zero\_conv}(z'_\theta(z_t + \text{zero\_conv}(c_{kw}), c, t)),\end{aligned} \tag{297}$$

where c_{kw} refers to the encoded keyword segment. During fine-tuning, we extract keyword segments from reference paraphrases, while for inference, we likewise harness the source input for keyword segments.

4 Experiment

4.1 Setups

Implementation Details LDP is implemented by N=12 layers of Transformer blocks with 12 attention heads, where we adopt time step embedding in the same way as the position embedding. The maximum sequence length is 96. We adopt *BART-base* (139M) for encoder and decoder, and the diffusion dimension is 768, the same as BART embedding dimension.

The diffusion process is defined by $T = 1000$, with cosine schedule (Nichol and Dhariwal, 2021) for β_t. We train the model for up to 250k steps with batch size 192. The learning rate for the DPM training is 10^{-4}, while for the controller fine-tuning is 10^{-5}. We apply DPM-Solver++(Lu et al., 2022b) with 25 steps for diffusion sampling.

	QQP				Twitter			
	BLEU↑	PPL↓	div-4↑	iBScore↑	BLEU↑	PPL↓	div-4↑	iBScore↑
Transformer	30.36	214.98	58.01	33.53	31.06	299.47	58.42	48.11
BART-FT	33.73	189.23	57.20	33.94	**33.21**	324.14	58.15	47.10
T5-GPVAE	33.50	200.13	64.15	25.89	27.01	129.86	58.59	52.82
BART-CVAE	32.21	194.81	55.40	47.02	32.24	325.71	59.77	44.37
DiffuSeq	24.13	397.68	**86.41**	49.11	9.83	1045.88	85.45	50.99
SeqDiffuSeq	24.32	404.28	70.35	48.28	11.92	903.26	71.51	52.46
LDP(ours)	36.56	267.53	73.22	50.57	18.75	466.46	**93.32**	59.72
LDP(ours) w/ES	**37.48**	246.76	73.63	**51.44**	19.72	419.63	93.26	**60.36**

Table 1: Results on paraphrase generation. Baselines are implemented from the source code. The bold indicates the top results, whereas the second best is underlined. LDP outperforms SOTA diffusion baselines. Overall, LDP with enforced semantics (LDP w/ES) achieves the best iBScore.

We trained our model on 4× v100 and 4× Nvidia 3090 GPUs for about 100 GPU-hours. LDP has approximately 200M trainable parameters.

Datasets We adopt Quora Question Pairs (QQP)[1], and Twitter-URL (Twitter) (Lan et al., 2017), which is popular amongst mainstream paraphrase generations. The QQP data is mass-extracted from Quora regarding the shared utterance, while the Twitter data is annotated from social media by semantic similarity. We randomly divide both datasets into three parts: 10k test sentences, 10k validation sentences, and the remaining sentences were assigned to the training set.

Baselines We first choose several mainstream paraphrase generations as baselines, including the deterministic paradigm by generic Transformer (Vaswani et al., 2017), fine-tuned BART-base (BART-FT) (Lewis et al., 2019), and variational paradigm by T5-GPVAE (Du et al., 2022) and BART-CVAE (Wang and Wan, 2019). We also include state-of-the-art text generations via the diffusion model, such as DiffuSeq (Gong et al., 2022) and SeqDiffuSeq (Yuan et al., 2022), which are versatile end-to-end diffusion modeling. DiffuSeq applies minimum Bayes risk decoding (Kumar and Byrne, 2004, MBR) with 10 candidates, while SeqDiffuSeq implements beam search for decoding given an on-the-fly encoder-decoder framework during rounding. For all beam search in the experiments, we apply beam size as 4.

Metrics An ideal paraphrase must achieve both generation quality and diversity. We validate text quality by reference-oriented BLEU (Papineni et al., 2002) and perplexity (PPL) based on GPT2 (Radford et al., 2019); The diversity is measured by intra-diversity within each generated sentence, where we adopt distinct unigram div-4 (Deshpande et al., 2019). Note that reference-oriented metrics such as BLEU contradict the intuition of generation diversity, we finally adopt iBScore (Dou et al., 2022) for *overall metric*, which measures the semantics by cosine similarity of BERT sentence embeddings, namely, BERTScore (Zhang* et al., 2020), and punishes source duplication by source BLEU:

$$\text{iBScore} = \text{BERTScore} - \text{srcBLEU},$$

where we calculate BERTScore via RoBERTa-large (Liu et al., 2019).

4.2 Main Results

Table 1 presents the main results of our experiments. LDP achieves comparable or even superior performance compared to mainstream baselines. Moreover, it outperforms other diffusion counterparts on QQP and Twitter test sets. LDP achieves the best performance amongst all baselines for the QQP test set.

Specifically, LDP improves overall results (iBScore) especially diversity (div-4) compared to traditional end-to-end paradigms such as fine-tuned BART, where we consider the diffusion module significantly improves the generation diversity. On the other hand, LDP also outperforms SOTA diffusion baselines such as SeqDiffuSeq in terms of BLEU and perplexity, where we consider the LDP to implement a better bridge between the diffusion process with the discrete texts than rounding. Additionally, unlike the intuition to introduce external features, the original case output improves greatly when we enforce the generation semantics with mere a segment of source inputs as shown in Table 2.

[1]https://www.kaggle.com/c/ quora-question-pairs

src	how is black money gon na go off with no longer the use of same 500 and 1000 notes ?
origin paraphrase	how does black money brought out to black money market or corruption?
enforcement	<M> <M> black money <M> <M> <M> <M> <M> no longer <M> <M> <M> <M> <M> <M> 1000 <M> <M>
enforced parphrase	how does banning 500 and 1000 rupee notes solve black money problem?

Table 2: Enforce paraphrase semantics by input segments. We inject input segments masked by placeholder <M> of 'black money', 'no longer', and '1000' via controller, which improves the paraphrase semantic.

Source	what should i do to improve my tennis?
DiffuSeq	what is the best way to improve your tennis **andtiv** month? what should i do to improve my tennis? how can i increase tennis? how can i improve my tennis? how do i improve my tennis skills?
BART -CVAE	how can i improve tennis skills? how can i get better in tennis? how do i improve my tennis? how do i improve tennis playing? how can i improve tennis skills?
LDP	what is the best way to be good at tennis? what are the best ways to get better at tennis? how can i improve to get better at tennis? how can i improve my skills at professional tennis? how can i improve my skill for tennis playing?

Table 3: LDP generates more fluent and diverse paraphrases compared to baselines. DiffuSeq even generates errors like '**andtiv**'.

Overall, LDP achieves the best iBScore, indicating a high-quality and diverse paraphrase generation. As shown in Table 3, we generate several paraphrases by different latent sampling for generations. Though BART-CVAE and DiffuSeq are SOTA generators for diversity, they still yield relatively resemble paraphrases. LDP, on the other hand, yields better and more diverse paraphrases.

	QQP	
Model	BLEU↑	BERTScore↑
DiffuSIA(Tan et al., 2023)	24.95	83.62
BG-DiffuSeq(Tang et al., 2023)	26.27	-
TESS(Mahabadi et al., 2023)	<u>30.2</u>	<u>85.7</u>
Dinoiser(Ye et al., 2023)	26.07	-
Diff-Glat(Qian et al., 2022)	29.86	-
SeqDiffuSeq(Yuan et al., 2022)	24.32	-
DiffuSeq(Gong et al., 2022)	24.13	83.65
LDP(ours)	**36.56**	**87.51**

Table 4: Results on QQP dataset compared with more diffusion generation baselines

We additionally include more recent text diffusion generators for comparison for QQP tests. Table 4 shows that LDP achieves state-of-the-art BLEU and BERTScore amongst the diffusion baselines.

5 Analysis

5.1 Inference Efficiency

	Timelapse (s)	Acceleration
Transformer	235	206.3×
BART-FT	238	203.7×
T5-GPVAE	3909	12.4×
BART-CVAE	252	192.4×
DiffuSeq(DDIM 2000)	48480	1×
DiffuSeq(DDIM 500)	11530	4.2×
SeqDiffuSeq(DDIM 2000)	13851	3.5×
LDP(ours)	290	**167.2**×

Table 5: Inference overhead on QQP validation set (seconds), where we set the DiffuSeq as the efficiency baseline.

By eliminating the rounding process in text diffusion models, LDP achieves improved efficiency. We compare several baseline efficiencies under their best-generation performance. As shown in Table 5, LDP is 167.2 times faster than DiffuSeq and 50 times faster than SeqDiffuSeq, where the diffusion overhead takes up only 18% of the total timelapse. Consequently, our approach achieves a similar efficiency compared to that of the autoregressive baselines with only encoder-decoder overheads.

Note that the rounding takes up considerable computation resources for minimum Bayes risk decoding, which limits the maximum dimension supported by diffusion. Therefore, DiffuSeq is modeled by only 128 dimensions. However, LDP by 768 dimensions is still 10.4× faster than DiffuSeq in a single-step text diffusion, excluding the impact of the sampling acceleration. Thus, the removal of the rounding, thereby eliminating the expensive quality insurance like MBR, contributes the efficiency with additional model dimensions for generation quality.

5.2 Domain Adaptation by Controller

The semantic controller is also apt for domain adaptation. We additionally adopt ChatGPT-augmented paraphrase dataset (Vladimir Vorobev, 2023)(ChatGPT-Aug) as the novel data domain, then sample 10k sentences as the test set. ChatGPT-

	QQP BLEU↑	ChatGPT-Aug BLEU↑
origin	36.56	10.16
+ES	36.46	14.65 (+4.49)

Table 6: Domain adaptation by controller

Aug dataset consists of paraphrases generated by ChatGPT (OpenAI, 2022) from an ensembled dataset including QQP, SQuAD 2.0 (Rajpurkar et al., 2016) and CNN news (See et al., 2017).

We first train the LDP on QQP data for 200k steps, then fine-tune it with a controller for ChatGPT-Aug data for 40k steps. The fine-tuning follows the same routine for controller training, where the ChatGPT-Aug data pairs are training sources and references, with additional keywords from its reference as controller inputs. Table 6 shows that adaptation by controller is viable, where the fine-tuned controller substantially improves ChatGPT-Aug test BLEU, with only minor loss for the original test domain. Note that, the performance of the original data domain is retained by inference without controller inputs, which outstands from traditional data adaptation by fine-tuning.

5.3 Ablation Study for Samplers

Model	QQP BLEU↑	PPL↓	div-4↑	iBScore↑
DiffuSeq(DDIM 2000)	24.13	397.68	86.41	49.11
DiffuSeq(DDIM 500)	0.17	1195.42	79.36	52.61
SeqDiffuSeq(DDIM 2000)	24.32	404.28	70.35	48.28
LDP(DDIM 500)	35.97	245.26	74.16	50.35
LDP(DPM-solver 25)	36.56	267.53	73.22	50.57

Table 7: Ablation study for diffusion samplers.

The diffusion sampler plays an important role during inference, thus we conduct an ablation study on the diffusion sampler for comparison. Intuitively, the diffusion generation improves by more steps given the same sampler. Due to the truncation errors introduced by rounding, ODE-based samplers are not apt for text diffusion with rounding, such as DiffuSeq and SeqDiffuSeq. Thus, we additionally adopt DDIM sampler (Nichol and Dhariwal, 2021) with 500 and 200 sampling steps, which is also adopted for diffusion baselines. Table 7 shows that LDP still outperforms the DiffuSeq and SeqDiffuSeq given the same sampler setting. Notably, LDP still outperforms the baselines by fewer sampling steps, which were expected to be inferior.

Model	Quasar-T BLEU↑	Quasar-T BERTScore↑
DiffuSIA	17.12	62.19
BG-DiffuSeq	17.53	-
TESS	19.50	65.8
SeqDiffuSeq	17.5	-
DiffuSeq	17.31	61.23
LDP(ours)	18.77	73.23
+ES	**19.50**	**73.40**

Table 8: Results on Quasar-T compared to other diffusion model baselines

5.4 Question Generation Ability

Considering the compatibility of LDP besides paraphrasing, we additionally validate question generation, which aims to generate the exact question by given issue descriptions. Question generation focuses on semantics upon relatively flexible utterance, thus we evaluate the BLEU and BERTScore on Quasar-T tests (Dhingra et al., 2017).

As shown in Table 8, LDP with enforced semantics also achieves a comparable BLEU score with SOTA, TESS (Mahabadi et al., 2023), with much improved BERTScore.

5.5 Diffusion Process during Generation

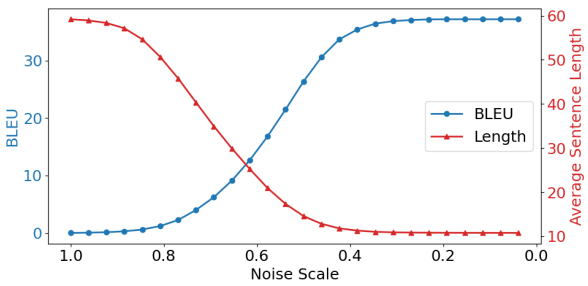

Figure 3: BLEU variance with averaged sentence length during LDP generation. The generation proceeds by diminishing noise scale with increasing BLEU and diminishing generation length.

We track the diffusion sampling during the LDP generation by our implementation in Figure 3, where we force-decode the intermediate latent representations as text. Intuitively, the BLEU score increases along the sampling process, whereas the generation length will decrease from the maximum to the actual target length. We present a generation process in Table 9, where the output semantics drastically morph towards desiderata during the middle of the generation process, with only minor modifications within a smaller noise scale, as indicated by Figure 3.

source	What is the best way to lose weight without diet
noise scale	outputs
1.0	This is the first time since 2009 that he has not been on the losing side. He has been on and off since then. He is now on the winning side. He is also on the receiving end. He was on the way home from the hospital. He was on his way home when he was
0.8	In the meantime, I have a feeling that this is going to be an interesting experience. The fact that I have the ability to do this shows that I'm not the only one. I'm the one who has the guts to do it. And I'm also the one that has the will to do
0.6	however, it is not the end of the world. It is the beginning of a new year. The end of a period of time.... The end of an era. The beginning of the year. The end to the period. The start of the next phase. The first phase.
0.4	how can i i lose weight? how can I lose weight? how do i? what can i lose? how did i? how do I? what do i eat?
0.2	how can i lose weight without doing exercise or diet??
0.0	how can i lose weight without doing exercise or diet?

Table 9: The generation process tracked by diffusion noise scale. The generation improves along the diffusion, with diminishing output length from the maximum.

6 Related Work

Paraphrase generation is commonly modeled by end-to-end sequence generation, such as fine-tuned Transformers (Vaswani et al., 2017; Lewis et al., 2019; Yang et al., 2019). However, such deterministic generation fails to ensure diversity, thus some researchers (Xie et al., 2022; Sancheti et al., 2022; Vijayakumar et al., 2016; Fan et al., 2018) introduce external features to amend. Others turn to the variational generation for diversity (Bowman et al., 2015). To amend the inferior generation by latent representation, some researchers further introduce pre-trained text encoder and decoder for generation (Du et al., 2022; Wang and Wan, 2019).

On the other hand, the diffusion model has been a popular Markov variational generation in recent years. Existing text diffusions cater to the discrete nature of language for versatile generations. DiffuSeq (Gong et al., 2022) employs a single Transformer encoder and partial noising process to extend Diffusion-LM (Li et al., 2022) for end-to-end text generations. They introduce discrete sampling for diffusion intervals called 'rounding', to ensure generation quality. However, rounding introduces truncation errors that hinder efficient skip-step diffusion sampler. BG-Diffuseq (Tang et al., 2023) aims to narrow the gap between training and sampling for text diffusion via incorporating distance penalty and adaptive decay sampling. Dinoiser (Ye et al., 2023), turns to mitigate truncation errors by manipulating the noise scale scheduled for text diffusion training and inference to improve its efficacy. On the other hand, SeqDiffuSeq (Yuan et al., 2022) turns to a continuous diffusion modeling within an on-the-fly encoder-decoder framework. Similarly, DiffuSIA (Tan et al., 2023) introduces an encoder-decoder diffusion via spiral interaction. Diff-GLAT (Qian et al., 2022) incorporates residual glancing sampling, a text reconstruction via encoder-decoder, with its dropout rate as the noise scale. TESS (Mahabadi et al., 2023) proposes a logit simplex space rather than embedding space for text diffusion.

7 Conclusion

We propose a novel paraphrase generation by the controllable latent diffusion model, LDP, which can further enforce semantics for paraphrase generation by harnessing only input segments instead of external features. Experiments show that LDP generates better paraphrase with superior efficiency compared to its diffusion counterparts. Further analysis shows that LDP is also applicable to other similar text generations such as question generation. Its controller is also helpful for domain adaptation.

Overall, LDP strikes a better balance between generation quality and diversity compared to mainstream baselines.

Limitations

In this work, we opt not to implement a larger Pre-trained model than the BART-base encoder and decoder, which will require more GPU memory during training. The latent diffusion is viable for the diffusion process by lower dimensions to cut down training expenses. However, the diffusion process needs to cater to the dimension of the Pre-trained latent space. Thus, this work does not explore the impact of model scaling.

We analyse our method mainly on QQP due to the restriction of the open-sourced baselines. Additionally, we are unable to explore more controller

implementations other than domain adaptation due to time constraints.

Ethics Statement

The authors declare no competing interests. The datasets used in the training and evaluation come from publicly available sources and do not contain sensitive content such as personal information.

References

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. 2021. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993.

Yu Bao, Hao Zhou, Shujian Huang, Lei Li, Lili Mou, Olga Vechtomova, Xinyu Dai, and Jiajun Chen. 2019. Generating sentences from disentangled syntactic and semantic spaces. *arXiv preprint arXiv:1907.05789*.

Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. 2015. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*.

Christian Buck, Jannis Bulian, Massimiliano Ciaramita, Wojciech Gajewski, Andrea Gesmundo, Neil Houlsby, and Wei Wang. 2017. Ask the right questions: Active question reformulation with reinforcement learning. *arXiv preprint arXiv:1705.07830*.

Mingda Chen, Qingming Tang, Sam Wiseman, and Kevin Gimpel. 2019. Controllable paraphrase generation with a syntactic exemplar. *arXiv preprint arXiv:1906.00565*.

Yi Chen, Haiyun Jiang, Lemao Liu, Rui Wang, Shuming Shi, and Ruifeng Xu. 2022. Mcpg: A flexible multi-level controllable framework for unsupervised paraphrase generation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5948–5958.

Aditya Deshpande, Jyoti Aneja, Liwei Wang, Alexander G Schwing, and David Forsyth. 2019. Fast, diverse and accurate image captioning guided by part-of-speech. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10695–10704.

Bhuwan Dhingra, Kathryn Mazaitis, and William W Cohen. 2017. Quasar: Datasets for question answering by search and reading. *arXiv preprint arXiv:1707.03904*.

Li Dong, Jonathan Mallinson, Siva Reddy, and Mirella Lapata. 2017. Learning to paraphrase for question answering. *arXiv preprint arXiv:1708.06022*.

Yao Dou, Chao Jiang, and Wei Xu. 2022. Improving large-scale paraphrase acquisition and generation. *arXiv preprint arXiv:2210.03235*.

Wanyu Du, Jianqiao Zhao, Liwei Wang, and Yangfeng Ji. 2022. Diverse text generation via variational encoder-decoder models with gaussian process priors. *arXiv preprint arXiv:2204.01227*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*.

Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and LingPeng Kong. 2022. Diffuseq: Sequence to sequence text generation with diffusion models. *arXiv preprint arXiv:2210.08933*.

Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. 2023. Latent video diffusion models for high-fidelity long video generation.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.

Tom Hosking, Hao Tang, and Mirella Lapata. 2022. Hierarchical sketch induction for paraphrase generation. *arXiv preprint arXiv:2203.03463*.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. 2022. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35:26565–26577.

Ashutosh Kumar, Satwik Bhattamishra, Manik Bhandari, and Partha Talukdar. 2019. Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3609–3619.

Shankar Kumar and Bill Byrne. 2004. Minimum bayes-risk decoding for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 169–176.

Zeqiang Lai, Xizhou Zhu, Jifeng Dai, Yu Qiao, and Wenhai Wang. 2023. Mini-dalle3: Interactive text to image by prompting large language models.

Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. 2017. A continuously growing dataset of sentential paraphrases. In *Proceedings of The 2017 Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pages 1235–1245. Association for Computational Linguistics.

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

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. 2022. Diffusion-lm improves controllable text generation. In *Advances in Neural Information Processing Systems*, volume 35, pages 4328–4343. Curran Associates, Inc.

Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D. Plumbley. 2023. Audioldm: Text-to-audio generation with latent diffusion models.

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

Justin Lovelace, Varsha Kishore, Chao Wan, Eliot Shekhtman, and Kilian Weinberger. 2022. Latent diffusion for language generation. *arXiv preprint arXiv:2212.09462*.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. 2022a. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. 2022b. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*.

Rabeeh Karimi Mahabadi, Jaesung Tae, Hamish Ivison, James Henderson, Iz Beltagy, Matthew E Peters, and Arman Cohan. 2023. Tess: Text-to-text self-conditioned simplex diffusion. *arXiv preprint arXiv:2305.08379*.

Alexander Quinn Nichol and Prafulla Dhariwal. 2021. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR.

OpenAI. 2022. https://openai.com/blog/chatgpt.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

William Peebles and Saining Xie. 2022. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*.

Lihua Qian, Mingxuan Wang, Yang Liu, and Hao Zhou. 2022. Diff-glat: Diffusion glancing transformer for parallel sequence to sequence learning. *arXiv preprint arXiv:2212.10240*.

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.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. *arXiv e-prints*, page arXiv:1606.05250.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Abhilasha Sancheti, Balaji Vasan Srinivasan, and Rachel Rudinger. 2022. Entailment relation aware paraphrase generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 10, pages 11258–11266.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Noam Shazeer. 2020. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*.

Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020a. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2020b. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.

Jiao Sun, Xuezhe Ma, and Nanyun Peng. 2021. Aesop: Paraphrase generation with adaptive syntactic control. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 5176–5189.

Chao-Hong Tan, Jia-Chen Gu, and Zhen-Hua Ling. 2023. Diffusia: A spiral interaction architecture for encoder-decoder text diffusion. *arXiv preprint arXiv:2305.11517*.

Zecheng Tang, Pinzheng Wang, Keyan Zhou, Juntao Li, Ziqiang Cao, and Min Zhang. 2023. Can diffusion model achieve better performance in text generation? bridging the gap between training and inference! *arXiv preprint arXiv:2305.04465*.

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

Ashwin K Vijayakumar, Michael Cogswell, Ramprasath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2016. Diverse beam search: Decoding diverse solutions from neural sequence models. *arXiv preprint arXiv:1610.02424*.

Maxim Kuznetsov Vladimir Vorobev. 2023. Chatgpt paraphrases dataset.

Tianming Wang and Xiaojun Wan. 2019. T-cvae: Transformer-based conditioned variational autoencoder for story completion. In *IJCAI*, pages 5233–5239.

Xuhang Xie, Xuesong Lu, and Bei Chen. 2022. Multi-task learning for paraphrase generation with keyword and part-of-speech reconstruction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1234–1243.

Jingjing Xu, Xu Sun, Zhiyuan Zhang, Guangxiang Zhao, and Junyang Lin. 2019. Understanding and improving layer normalization. *CoRR*, abs/1911.07013.

Haoran Yang, Wai Lam, and Piji Li. 2021. Contrastive representation learning for exemplar-guided paraphrase generation. *arXiv preprint arXiv:2109.01484*.

Kexin Yang, Dayiheng Liu, Wenqiang Lei, Baosong Yang, Haibo Zhang, Xue Zhao, Wenqing Yao, and Boxing Chen. 2022. Gcpg: A general framework for controllable paraphrase generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 4035–4047.

Qian Yang, Zhouyuan Huo, Dinghan Shen, Yong Cheng, Wenlin Wang, Guoyin Wang, and Lawrence Carin. 2019. An end-to-end generative architecture for paraphrase generation. 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 3132–3142.

Jiasheng Ye, Zaixiang Zheng, Yu Bao, Lihua Qian, and Mingxuan Wang. 2023. Dinoiser: Diffused conditional sequence learning by manipulating noises. *arXiv preprint arXiv:2302.10025*.

Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Fei Huang, and Songfang Huang. 2022. Seqdiffuseq: Text diffusion with encoder-decoder transformers. *arXiv preprint arXiv:2212.10325*.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. 2023. Adding conditional control to text-to-image diffusion models.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Jianing Zhou and Suma Bhat. 2021. Paraphrase generation: A survey of the state of the art. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 5075–5086.