

# TESS: Text-to-Text Self-Conditioned Simplex Diffusion

Rabeeh Karimi Mahabadi<sup>1,4\*</sup> Hamish Ivison<sup>3,5\*†</sup> Jaesung Tae<sup>2</sup>  
James Henderson<sup>4</sup> Iz Beltagy<sup>3</sup> Matthew E. Peters<sup>3†‡</sup> Arman Cohan<sup>2,3‡</sup>

<sup>1</sup>EPFL <sup>2</sup>Yale University <sup>3</sup>Allen Institute for AI  
<sup>4</sup>Idiap Research Institute <sup>5</sup>University of Washington  
rabeeh.karimimahabadi@epfl.ch, hamishi@allenai.org

## Abstract

Diffusion models have emerged as a powerful paradigm for generation, obtaining strong performance in various continuous domains. However, applying continuous diffusion models to natural language remains challenging due to its discrete nature and the need for a large number of diffusion steps to generate text, making diffusion-based generation expensive. In this work, we propose Text-to-text Self-conditioned Simplex Diffusion (TESS), a text diffusion model that is fully non-autoregressive, employs a new form of self-conditioning, and applies the diffusion process on the logit simplex space rather than the learned embedding space. Through extensive experiments on natural language understanding and generation tasks including summarization, text simplification, paraphrase generation, and question generation, we demonstrate that TESS outperforms state-of-the-art non-autoregressive models, requires fewer diffusion steps with minimal drop in performance, and is competitive with pre-trained autoregressive sequence-to-sequence models. We publicly release our codebase.<sup>1</sup>

## 1 Introduction

Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021) have achieved state-of-the-art performance in various continuous domains, such as image (Nichol and Dhariwal, 2021), audio (Kong et al., 2020; Shen et al., 2023), video (Ho et al., 2022), and text-to-image generation (Saharia et al., 2022; Ramesh et al., 2022). Inspired by the success of diffusion for continuous domains, recent works have adapted diffusion to discrete spaces, such as text (Austin et al., 2021; Hoogeboom et al., 2021; Savinov et al., 2021; Reid et al., 2022). One line of work proposes diffusing

the model latent space by adding Gaussian noise to input word embeddings (Li et al., 2022b). Another approach, SSD-LM (Han et al., 2022), adds noise to the vocabulary probability simplex.

Direct diffusion on the probability simplex is desirable (Richemond et al., 2022) as it eliminates the need for an extra step to map diffused embeddings to actual discrete inputs or auxiliary methods such as binary encoding (Chen et al., 2022). Despite its strong performance, however, SSD-LM has several shortcomings: a lack of self-conditioning (Chen et al., 2022), a lack of extensive evaluation on downstream tasks, and most notably, its restriction to generating blocks of 25 tokens, which hinders the potential benefits of full diffusion, e.g., the ability to perform arbitrary infilling, flexible generation, and a global view of the sequence.

In this work, we present TESS, a text-to-text diffusion model, which overcomes several limitations of prior works: restrictions on scale (Hoogeboom et al., 2021; Austin et al., 2021), dependence on pretrained embeddings (Strudel et al., 2022), semi-autoregressive nature (Han et al., 2022), and short generation length (Gong et al., 2023). TESS closely follows Han et al. (2022, 2023a) by performing diffusion on the vocabulary logit space rather than the typical embedding space. Unlike SSD-LM, however, TESS is fully non-autoregressive and performs diffusion on the entire sequence. It also incorporates a novel form of self-conditioning, which demonstrates a competitive edge over the original self-conditioning method (Chen et al., 2022) and dramatically improves the efficiency and quality of TESS.

We evaluate TESS on a suite of natural language generation (NLG) tasks including summarization, text simplification, paraphrase generation, and question generation. Our empirical results surpass the current state-of-the-art non-autoregressive and diffusion-based approaches and are on par with a strong pretrained encoder-decoder language

\* Co-first authors.

† Work done during employment at AI2.

‡ Equal advising.

<sup>1</sup><https://github.com/allenai/tess-diffusion>

model (Lewis et al., 2020). In particular, our simplex-based self-conditioning method substantially improves generation quality. We also evaluate TESS on natural language understanding (NLU) tasks from the GLUE benchmark (Wang et al., 2019) and show that it performs comparably to strong masked language model baselines. Our contributions can be summarized as follows.

1. We demonstrate the effectiveness of a fully non-autoregressive scheme for text diffusion models, which outperforms strong autoregressive and non-autoregressive baselines.
2. We propose a new self-conditioning method that exploits the simplex semantics of the diffusion space and greatly improves performance.
3. We evaluate TESS on a suite of diverse NLG and NLU tasks, highlighting the effectiveness of our text-to-text simplex diffusion paradigm.
4. We show TESS’ fully non-autoregressive approach results in faster and more efficient sampling than semi and fully autoregressive methods for long sequences.

We will release our trained models and code to promote open research in the field of diffusion-based text generation.

## 2 Background

We revisit continuous diffusion models (Sohl-Dickstein et al., 2015), following the formulation of Denoising Diffusion Models (Ho et al., 2020; Song et al., 2020).

**Training** Given a sample  $\mathbf{x}_0 \in \mathbb{R}^d$  from a data distribution  $p_{\text{data}}$ , a forward diffusion process  $q(\mathbf{x}_t|\mathbf{x}_{t-1})$  is a Markov chain that generates a sequence of latent variables  $\mathbf{x}_1, \dots, \mathbf{x}_T$  by gradually adding Gaussian noise at each time step  $t \in \{1, 2, \dots, T\}$  with variance  $\beta_t \in \mathbb{R}_{>0}$ :

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1 - \beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}). \quad (1)$$

Let  $\epsilon_t \sim \mathcal{N}(0, \mathbf{I})$ ,  $\alpha_t = 1 - \beta_t$ , and  $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$ . Then sampling  $\mathbf{x}_t$  at an arbitrary time step  $t$  has the closed-form solution

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon_t. \quad (2)$$

Given a well-behaved noise schedule  $\{\beta_t\}_{t=1}^T$ ,  $\mathbf{x}_T$  follows a stationary prior distribution  $\mathcal{N}(0, \mathbf{I})$ . Therefore, if we can approximate the reverse process  $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$  via a model  $p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)$  with

parameters  $\theta$ , then we can sample random noise from a standard Gaussian and gradually denoise it to sample from  $p_{\text{data}}$ . In our settings, our model  $p_{\theta}$  is a transformer model<sup>2</sup>. The reverse process is thus parametrized as

$$p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\boldsymbol{\mu}_{\theta}(\mathbf{x}_t, t), \boldsymbol{\Sigma}_{\theta}(\mathbf{x}_t, t)). \quad (3)$$

The model is trained by minimizing the mean squared error between the ground-truth data  $\mathbf{x}_0$  and its estimate  $\hat{\mathbf{x}}_{\theta}$ :<sup>3</sup>

$$\mathcal{L} = \mathbb{E}_{t, q(\mathbf{x}_0), q(\mathbf{x}_t|\mathbf{x}_0)} \|\mathbf{x}_0 - \hat{\mathbf{x}}_{\theta}(\mathbf{x}_t, t)\|^2. \quad (4)$$

**Noise schedule** The forward diffusion process is defined by a noise schedule. In this work, we follow the cosine schedule (Nichol and Dhariwal, 2021) for  $\alpha_t$ :

$$\bar{\alpha}_t = \frac{f(t)}{f(0)}, \quad f(t) = \cos\left(\frac{t/T + s}{1 + s} \cdot \frac{\pi}{2}\right)^2. \quad (5)$$

**Inference** In Song et al. (2020), model predictions are iteratively denoised for  $t = T, \dots, 1$  starting from pure noise, following

$$\mathbf{x}_{t-1} = \sqrt{\alpha_{t-1}}\hat{\mathbf{x}}_{\theta} + \sqrt{1 - \alpha_{t-1}} \cdot \frac{\mathbf{x}_t - \sqrt{\alpha_t}\hat{\mathbf{x}}_{\theta}}{\sqrt{1 - \alpha_t}}.$$

We follow the recently proposed simplex-based diffusion procedure by Han et al. (2022), which allows us to apply diffusion to text without employing auxiliary methods that map categorical data to continuous space (Richemond et al., 2022).

## 3 Method

In this section, we present TESS, a simplex diffusion-based text-to-text model. Building upon SSD-LM (Han et al., 2022), we propose a fully non-autoregressive model with self-conditioning.

**Continuous data representation** Let  $\mathcal{V}$  denote the vocabulary space. Following Han et al. (2022), we map the ID of each token to be generated  $w \in \mathcal{V}$  to a  $k$ -logit simplex to produce  $\mathbf{s}^w \in \{\pm k\}^{|\mathcal{V}|}$ , whose  $i$ -th component satisfies

$$s_{(i)}^w = \begin{cases} k, & \text{if } i = w, \\ -k, & \text{otherwise,} \end{cases} \quad (6)$$

<sup>2</sup>Specifically, we use a RoBERTa model (Liu et al., 2019), but our formulation could be applied to any transformer variant.

<sup>3</sup>Alternatively, we can train the model to predict the added noise; see Ho et al. (2020). See also Song et al. (2021) for a score-matching interpretation.

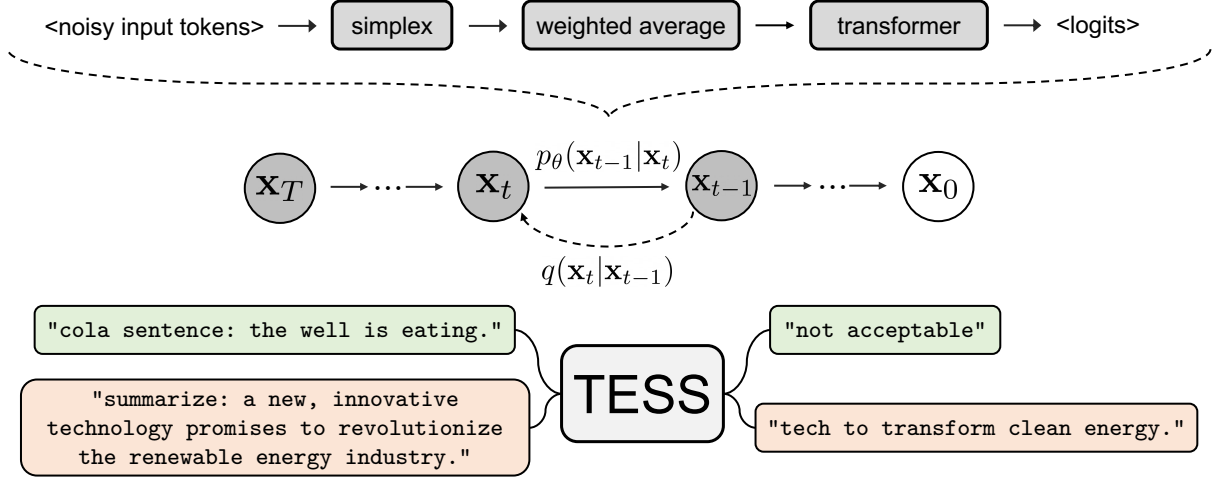


Figure 1: Overview of TESS. During training (top), we first add noise to the vocabulary probability simplex, compute a weighted average word embedding, and denoise it using a transformer encoder. To generate from our model, we begin with noise and iteratively refine it into a final logit distribution (middle). The resulting model can be used for a wide range of NLG and NLU end tasks (bottom).

with a hyperparameter  $k \in \mathbb{R}$ . We then produce a probability simplex over  $\mathcal{V}$  via  $\mathbf{p}^w = \text{softmax}(\mathbf{s}^w)$ . Finally, we compute the weighted sum of word embeddings to obtain a continuous embedding vector,  $\mathbf{h}^w = \mathbf{E}\mathbf{p}^w$ , where  $\mathbf{E} \in \mathbb{R}^{d \times |\mathcal{V}|}$  is the word embedding matrix,  $d$  denotes the size of the hidden dimension, and  $\mathbf{h}^w \in \mathbb{R}^d$ .

**Time step embeddings** After computing the continuous word embeddings, we add the time step embeddings to inform the model of the current time step. Our time step embedding is a linear layer, and we feed scaled time steps  $t/T$  to this layer. The output is a time step embedding in  $\mathbb{R}^d$  that is added to  $\mathbf{h}_w$  to produce the final latent input vector.

**Text-to-text non-autoregressive modeling** Unlike SSD-LM, which feeds small blocks of text to semi-autoregressively generate sequences of text, we feed the entire latent vector along with the context into an encoder transformer model. This is a key difference between our approach and SSD-LM, as it allows for a fully non-autoregressive model capable of generating sequences of any length. In practice, our evaluation tasks often require output sequences of 100 tokens or more, and by moving to a fully non-autoregressive paradigm, we are able to generate entire output sequences in parallel without resorting to semi-autoregressive generation.

**Forward diffusion** Let  $\mathbf{w} = (w_1, \dots, w_L)$  be a sentence of  $L$  tokens such that  $w_i \in \mathcal{V}$ , and  $\mathbf{S}_0 = (\mathbf{s}^{w_1}, \dots, \mathbf{s}^{w_L}) \in \{\pm k\}^{L \times |\mathcal{V}|}$  be the  $k$ -logit simplex representation of  $\mathbf{w}$ . We add noise to the

$k$ -logit simplex representation during training according to

$$\mathbf{S}_t = \sqrt{\bar{\alpha}_t} \mathbf{S}_0 + \sqrt{1 - \bar{\alpha}_t} \boldsymbol{\epsilon}_t, \quad (7)$$

where subscript denotes the time step and  $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, k^2 \mathbf{I})$ .

**Training** Typical diffusion models are trained with mean squared error loss as in Equation (4) to predict the ground-truth data. This objective is known to be unstable for text diffusion models (Dieleman et al., 2022). Strudel et al. (2022) froze word embeddings and used specific scaling to deal with training instability. In this work, following Han et al. (2022), we instead compute the usual cross-entropy loss between the ground-truth tokens  $\mathbf{w}$  and the model prediction given a noisy logit simplex  $\mathbf{S}_t$  at time step  $t$ .

$$\mathcal{L} = \mathbb{E}_{t, q(\mathbf{S}_0), q(\mathbf{S}_t|\mathbf{S}_0)} \left[ - \sum_{i=1}^L \log p_\theta(w_i | \mathbf{S}_t, t) \right]. \quad (8)$$

**Sampling** During inference, we sample  $\mathbf{S}_T$  from the prior  $\mathcal{N}(0, k^2 \mathbf{I})$  and run the reverse process for  $t = T, \dots, 1$  on the noisy  $k$ -logit simplex. The reverse process can be approximated via

$$\mathbf{S}_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \hat{\mathbf{S}}_\theta(\mathbf{S}_t, t) + \sqrt{1 - \bar{\alpha}_{t-1}} \boldsymbol{\epsilon}_t. \quad (9)$$

See Appendix C for details. This resembles the forward process in Equation (7), which allows for an intuitive interpretation: to reverse one step from  $t$ ,

we take the model prediction  $\hat{\mathbf{S}}_\theta$  as the hypothetical ground-truth, then corrupt it by  $(t - 1)$  time steps. To construct the model prediction, we project the logits predicted by the underlying encoder model via  $\text{argmax}$  as a pseudo-inverse of Equation (6) to match the initial  $k$ -logit representation:

$$\hat{s}_{(i)}^w = \begin{cases} k, & \text{if } i = \text{argmax}(\mathbf{s}^w), \\ -k, & \text{otherwise.} \end{cases} \quad (10)$$

**Self-conditioning** In typical diffusion models, the model predicts the original data  $\mathbf{x}_0$  conditioned on its corrupted version, i.e.,  $\hat{\mathbf{x}}_0^t = \hat{\mathbf{x}}_\theta(\mathbf{x}_t, t)$ , where  $\hat{\mathbf{x}}_0^t$  denotes the estimate of  $\mathbf{x}_0$  at time step  $t$ . In this setting, the model’s estimates at previous time steps are discarded. However, in self-conditioning (Chen et al., 2022), the model conditions its prediction on both  $\mathbf{x}_t$  and its previously generated output, i.e.,  $\hat{\mathbf{x}}_0^t = \hat{\mathbf{x}}_\theta(\mathbf{x}_t, \hat{\mathbf{x}}_0^{t+1}, t)$ . To adapt the model for self-conditioning, we stochastically zero out the self-condition such that

$$\hat{\mathbf{x}}_0^t = \begin{cases} \hat{\mathbf{x}}_\theta(\mathbf{x}_t, \hat{\mathbf{x}}_0^{t+1}, t), & \text{with probability } \rho \\ \hat{\mathbf{x}}_\theta(\mathbf{x}_t, 0, t), & \text{otherwise,} \end{cases} \quad (11)$$

where the self-conditioning previous prediction is computed as  $\hat{\mathbf{x}}_0^{t+1} = \hat{\mathbf{x}}_\theta(\mathbf{x}_{t+1}, 0, t + 1)$ , with gradients detached. We set  $\rho = 0.5$  during training; during inference, we always use self-conditioning ( $\rho = 1$ ).

We propose a new self-conditioning method that exploits the simplex nature of our diffusion space. Let  $\mathbf{s}_t \in \mathbb{R}^{|V|}$  be a noised  $k$ -logit simplex for an arbitrary token  $w$ .<sup>4</sup> Instead of concatenating the previous prediction with  $\mathbf{s}_t$  and re-projecting, we first compute the average of simplex probabilities

$$\mathbf{p}_{\text{avg}}^w = \frac{1}{2} (\text{softmax}(\mathbf{s}_t) + \text{softmax}(\hat{\mathbf{s}}_0^{t+1})). \quad (12)$$

Note that  $\mathbf{p}_{\text{avg}}^w$  is a well-defined categorical distribution over  $V$ . We then compute a continuous embedding vector,  $\mathbf{h}^w = \mathbf{E}\mathbf{p}_{\text{avg}}^w$ , and use this vector as input to our underlying model to make a prediction for the given diffusion step following Equation 9. This is more efficient than the original self-conditioning method, which projects down the concatenated vectors. In Section §6.2, we also demonstrate the empirical effectiveness of this method over the original.

<sup>4</sup>We write  $\mathbf{s}_t^w$  as  $\mathbf{s}_t$  for brevity.

**Variable sequence length** A notable challenge in non-autoregressive generation is the assumption of fixed sequence lengths during inference. To overcome this issue, we follow prior work in embedding-space diffusion by using padding tokens (Li et al., 2022b). Specifically, during training, we always pad the variable-length output sequence to a fixed length using padding tokens. These padding tokens are included when computing the cross-entropy loss so that TESS learns to generate them. During inference, we specify the maximum sequence length and run sampling as usual.

## 4 Experiments

### 4.1 Tasks and Datasets

**Paraphrase generation** This task involves rephrasing a sentence while maintaining the semantics of the original. We use Quora Question Pairs (QQP),<sup>5</sup> which is composed of 147K positive pairs. We use only the positively-labelled pairs, which have the same meaning.

**Text simplification** This task involves simplifying complex sentences while retaining their original meaning. We use the NEWSLA-AUTO dataset (Jiang et al., 2020), which is composed of 666K complex-simplified sentences.

**Question generation** This task involves generating a question given an input context. We use the QUASAR-T dataset (Dhingra et al., 2017) processed by Yuan et al. (2022), resulting in 119K document-question pairs.

**Summarization** We evaluate our method on the CNN-DailyMail dataset (Hermann et al., 2015), which comprises 300K articles and summaries.

**Classification** We consider a set of classification tasks in the GLUE benchmark (Wang et al., 2019) covering a variety of tasks, including paraphrase detection (MRPC, QQP), sentiment classification (SST-2), natural language inference (MNLI,<sup>6</sup> RTE, QNLI), and linguistic acceptability (CoLA).<sup>7</sup>

### 4.2 Baselines

We compare TESS to several autoregressive baselines as well as state-of-the-art text diffusion mod-

<sup>5</sup><https://www.kaggle.com/c/quora-question-pairs>

<sup>6</sup>We report the accuracy on the matched validation set.

<sup>7</sup>Following Devlin et al. (2019); Raffel et al. (2020), as a common practice and due to the adversarial nature of WNLI, we do not experiment with WNLI.



Model	Paraphrase Generation			
	BLEU	BERT	R-L	D-1/4
<i>Autoregressive Models</i>				
BART (Lewis et al., 2020)	<u>30.4</u>	<u>85.7</u>	61.4	<u>98.8/61.0</u>
GPT-2 <sub>base</sub> <sup>†</sup> (Radford et al., 2019)	19.8	82.5	52.1	98.0/ <u>62.5</u>
GPT-2 <sub>large</sub> <sup>†</sup> (Radford et al., 2019)	20.6	83.6	54.2	98.2/50.2
GPVAE-T5 <sup>†</sup> (Du et al., 2022)	24.1	84.7	58.9	96.9/61.7
<i>Non-Autoregressive Models</i>				
LevT <sup>†</sup> (Gu et al., 2019)	22.7	83.4	57.9	97.9/33.3
<i>Non-Autoregressive Diffusion Models</i>				
DiffuSeq* (Gong et al., 2023)	18.5	79.5	—	97.6/—
SeqDiffuSeq* (Yuan et al., 2022)	23.3	82.9	—	98.1/—
SSD-LM (Han et al., 2022)	22.9	83.8	58.3	<b>98.8/57.3</b>
TESS (Ours)	<b>30.2</b>	<b>85.7</b>	<b>62.2</b>	98.5/ <b>61.1</b>

Table 1: Results on Paraphrase Generation task. <sup>†</sup> indicates results from from Gong et al. (2023), \* indicates results from Yuan et al. (2022). Boldfaced results show the best across all non-AR models; underlined results are the best across all models.

els. For autoregressive methods, we consider GPT-2 (Radford et al., 2019), BART (Lewis et al., 2020), and GPVAE-T5 (Du et al., 2022), a latent-structured variable model and an extension to T5 (Raffel et al., 2020). For text diffusion models, we consider Diffuser (Reid et al., 2022), DiffuSeq (Gong et al., 2023), SeqDiffuSeq (Yuan et al., 2022), SUNDAE (Savinov et al., 2021), LevT (Gu et al., 2019), a widely used iterative non-autoregressive model, and SSD-LM (Han et al., 2022) initialized from the same pretrained RoBERTa model as TESS and trained using the official SSD-LM codebase.<sup>8</sup> We report results without using additional decoding methods such as minimum Bayes risk decoding. We provide further details on baseline results in Appendix A.

### 4.3 Evaluation

For summarization, we report ROUGE-1 (R1), ROUGE-2 (R2), and ROUGE-L (R-L) variants (Lin, 2004) as done in prior text summarization work (Lewis et al., 2020). We quantify both generation quality and diversity. For evaluating generation quality, we report BLEU (Papineni et al., 2002), ROUGE-L (Lin, 2004) and BERTScore (Zhang et al., 2020) following Gong et al. (2023) and Yuan et al. (2022). For evaluating diversity, we report distant unigrams (D-1) and diverse 4-grams (D-4) (Deshpande et al., 2018). For text simplification, we use the standard SARI (Xu et al., 2016), and fol-

<sup>8</sup><https://github.com/xhan77/ssd-lm>

Model	Text Simplification			
	SARI	BLEU	BERT	R-L
<i>Autoregressive Models</i>				
BART (Lewis et al., 2020)	49.9	41.4	81.7	58.1
GPT-2 <sub>base</sub> <sup>†</sup> (Radford et al., 2019)	—	30.8	80.2	54.6
GPT-2 <sub>large</sub> <sup>†</sup> (Radford et al., 2019)	—	26.9	78.8	51.1
GPVAE-T5 <sup>†</sup> (Du et al., 2022)	—	33.9	81.7	58.3
<i>Non-Autoregressive Models</i>				
LevT <sup>†</sup> (Gu et al., 2019)	—	20.5	72.5	44.0
<i>Non-Autoregressive Diffusion Models</i>				
DiffuSeq* (Gong et al., 2023)	—	29.9	79.1	—
SeqDiffuSeq* (Yuan et al., 2022)	—	37.1	82.1	—
SSD-LM (Han et al., 2022)	36.3	12.5	69.5	39.6
TESS (Ours)	<b>54.3</b>	<b>41.5</b>	<b>82.1</b>	<b>59.4</b>

Table 2: Results on the text simplification task. <sup>†</sup> indicates results from from Gong et al. (2023), \* indicates results from Yuan et al. (2022).

lowing Gong et al. (2023); Yuan et al. (2022), we also include BLEU, BERTScore, and ROUGE-L.

## 4.4 Implementation

We start from the RoBERTa pretrained checkpoint (Liu et al., 2019) and finetune the model on downstream tasks using our proposed self-conditioned simplex diffusion method. The number of diffusion sampling steps at inference time is set to  $T = 1000$  for generation and  $T = 10$  for classification tasks. During training, we use  $T = 5000$ . We set the simplex scale to  $k = 5$ . Additional details are listed in Appendix A.

## 5 Results

### 5.1 Paraphrase Generation

As seen in Table 1, TESS significantly outperforms GPT-2 and other non-autoregressive and diffusion baselines in quality metrics (BLEU, BERT, and ROUGE) while achieving parity in diversity metrics (D-1/D-4). Moreover, TESS obtains competitive overall performance with BART. Note that BART uses a denoising pretraining objective, which is substantially conducive to sequence-to-sequence tasks (Lewis et al., 2020); we do not perform any additional pretraining beyond RoBERTa’s checkpoint, which was only pretrained on the general masked language modeling objective. We suspect that TESS could significantly benefit from additional diffusion pretraining (see Section §8).

Model	Question Generation			
	BLEU	BERT	R-L	D-1/4
<i>Autoregressive Models</i>				
BART (Lewis et al., 2020)	17.4	66.2	38.8	98.2/61.7
GPT-2 <sub>base</sub> <sup>†</sup> (Radford et al., 2019)	7.4	60.5	27.2	96.0/92.2
GPT-2 <sub>large</sub> <sup>†</sup> (Radford et al., 2019)	11.1	63.5	32.2	96.7/80.6
GPVAE-T5 <sup>†</sup> (Du et al., 2022)	12.5	63.1	33.9	93.8/72.8
<i>Non-Autoregressive Models</i>				
LevT <sup>†</sup> (Gu et al., 2019)	9.3	54.9	28.9	89.1/47.8
<i>Non-Autoregressive Diffusion Models</i>				
DiffuSeq* (Gong et al., 2023)	15.8	59.4	—	91.1/—
SeqDiffuSeq* (Yuan et al., 2022)	17.2	61.4	—	92.7/—
SSD-LM (Han et al., 2022)	14.1	62.8	38.5	94.5/56.9
TESS (random init)	19.0	60.8	36.1	96.1/62.4
TESS (Ours)	<b>19.5</b>	<b>65.8</b>	<b>38.9</b>	<b>97.1/63.0</b>

Table 3: Results on Question Generation task.

## 5.2 Text Simplification

Results of the text simplification task on the NEWSLA-AUTO dataset are presented in Table 2. TESS outperforms all baselines typically by large margins, including both autoregressive and non-autoregressive models.

## 5.3 Question Generation

As shown in Table 3, TESS outperforms other diffusion and non-autoregressive models in terms of both quality of generation (BLEU, BERTScore, ROUGE) and diversity (D-1/D-4). It also consistently outperforms other autoregressive baselines except for BART, whose performance is closely matched by TESS. We also train and evaluate TESS without initializing from pretrained RoBERTa (random init), and find that this outperforms all NAR baselines in BLEU and D-1, while remaining close in performance in BERTScore and ROUGE-L. This shows that the TESS framework outperforms baselines even without the benefit of making use of existing pretrained models.

## 5.4 Summarization

As shown in Table 4, TESS achieves competitive results with BART while outperforming prior diffusion work, Diffuser (Reid et al., 2022), by 1.9 ROUGE-L points, and its bootstrapped variants by 0.8 ROUGE-L points. Note that additional bootstrapping is orthogonal to their method and can be applied to TESS as well. Additionally, TESS outperforms GENIE, another prior diffusion-based method, while using half the number of diffusion steps. This suggests TESS’ simplex-based formulation leads to better performance than alternate

Model	CNN-DM		
	R1	R2	R-L
<i>Autoregressive Models</i>			
BART (Lewis et al., 2020)	42.9	20.1	40.1
Transformer (Vaswani et al., 2017) <sup>◊</sup>	—	—	36.8
<i>Non-Autoregressive Diffusion Models</i>			
SUNDAE (Savinov et al., 2021) <sup>◊</sup>	—	—	37.0
Diffuser (Reid et al., 2022) <sup>◊</sup>	—	—	37.8
Diffuser+ AR bootstrap <sup>◊</sup>	—	—	38.4
Diffuser + source bootstrap <sup>◊</sup>	—	—	38.9
GENIE (Lin et al., 2023)	41.8	18.3	35.5
TESS (Ours)	<b>42.3</b>	<b>19.4</b>	<b>39.7</b>

Table 4: Results on CNN-DailyMail dataset. Baseline values marked with <sup>◊</sup> are taken from Reid et al. (2022).

diffusion approaches.

## 5.5 Text Classification

To our knowledge, TESS is the first model that is evaluated on both NLG and NLU. We evaluate TESS on classification tasks and directly compare our diffusion-based finetuning method with standard finetuning methods for supervised learning. To perform a controlled experiment, we compare TESS with a similar-sized RoBERTa, which we use to initialize our model. Note that since TESS is text-to-text, similar to T5 (Raffel et al., 2020), it can naturally handle classification tasks by generating class labels without the need for verbalizers. For STS-B, which is a regression problem, we recast it as a 21-class classification problem following Raffel et al. (2020). Results are shown in Table 5. We observe that TESS matches or outperforms finetuned RoBERTa on several tasks, achieving roughly 2-point gains on MRPC and RTE.

## 6 Analysis

### 6.1 Variable Length Output

Figure 2 shows TESS is capable of producing outputs of variable lengths that match the underlying distribution of sequence lengths in the gold data as well as BART outputs. We also evaluate generation quality for differing output lengths in Figure 3. We observe that TESS is consistent with the BART baseline for variable target lengths, with longer generations matching BART’s performance.

### 6.2 Self-Conditioning

To examine the impact of self-conditioning, we compare our proposed method with the original strategy (Chen et al., 2022) in text simplification

Method	MNLI	QNLI	QQP	RTE	SST-2	MRPC	CoLA	STS-B	WNLI	Average
RoBERTa <sub>large</sub>	<b>90.2/90.2</b>	<b>94.7</b>	<b>92.2</b>	86.6	<b>96.4</b>	90.9	<b>68.0</b>	<b>92.4</b>	<b>91.3</b>	<b>88.9</b>
TESS <sub>large</sub> (Ours)	90.1/89.8	94.2	89.1	<b>88.5</b>	<b>96.4</b>	<b>93.1</b>	67.7	88.9	83.1	88.5

Table 5: Comparison of TESS and RoBERTa on GLUE tasks on the development set. Following Devlin et al. (2019), for MRPC and QQP, we report F1 score; STS-B, Spearman correlation coefficient; CoLA, Matthews correlation. For all other tasks, we report accuracy. Bold fonts indicate the best results.

Model	Text Simplification			
	SARI	BLEU	BERT	R-L
TESS	44.1	30.8	78.8	52.6
+orig. self-cond	52.2	<b>43.3</b>	81.9	58.9
+proposed self-cond	<b>54.2</b>	40.8	<b>82.0</b>	<b>59.3</b>
Model	Paraphrase generation			
	BLEU	BERT	R-L	D-1/D-4
TESS	25.9	84.4	59.7	<b>98.7/60.4</b>
+orig. self-cond	28.4	85.5	<b>61.7</b>	98.6/60.6
+proposed self-cond	<b>29.2</b>	<b>85.5</b>	61.2	98.5/ <b>61.4</b>

Table 6: Ablation on the effects of self-conditioning. We compare our proposed self-conditioning to the original method in Chen et al. (2022), and the model without self-conditioning. Bold fonts indicate the best results.

and paraphrase generation tasks. As shown in Table 6, adding self-conditioning consistently improves results, with our variant delivering the best overall performance.

### 6.3 Sampling Steps

We also investigate the quality of TESS generations on the suite of NLG tasks as well as MRPC by varying the number of sampling steps during inference. As shown in Table 7, TESS performs well with relatively few steps, with only a marginal drop in performance even when sampling steps are decreased from 1000 to 100. For classification tasks that involve shorter generation (MRPC), 10 sampling steps result in lossless quality. We also find that decreasing the number of sampling steps is possible in generative tasks like question generation. We provide results in Appendix B. Notably, it appears that the number of steps required correlates with the difficulty of the task: while classification tasks such as MRPC only require few steps, longer generation tasks such as CNN-DM require closer to 100 steps to achieve good performance.

### 6.4 Sampling Speed

We compare TESS generation speed with other models in Figure 4. We time how long decoding 25 to 5000 tokens takes given a context of 50 tokens

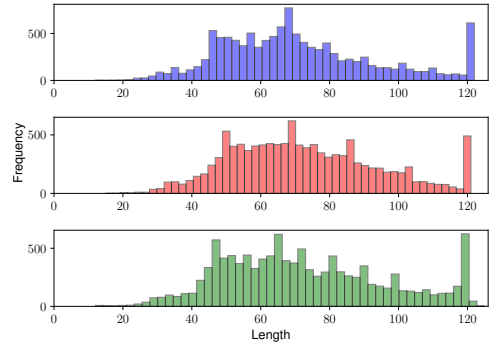


Figure 2: TESS is capable of producing output of variable lengths, matching target sequence lengths and BART outputs for CNN-DM generations. From top to bottom: (a) distribution of target lengths; (b) distribution of predicted length by BART; (c) distribution of predicted length by TESS.

and 100 diffusion steps. We find that TESS is substantially faster than SSD-LM, especially when SSD-LM has to generate multiple blocks due to its limited block size. Notably, we find TESS is faster, albeit marginally, than an equivalently-sized BART when generating more than 2000 tokens. We provide further details in Appendix A.5.

### 6.5 TESS vs other Diffusion Methods

As shown in Section 5, TESS outperforms other diffusion methods across several benchmarks. We believe this is due to a number of factors: (1) the simplex-based formulation being a more natural fit for language than embedding-based ones, allowing us avoid methods like clamping or an extra decoder for exiting the embedding space; (2) the simplex-based self-conditioning formulation, which we empirically show outperforms more standard self-conditioning methods (Table 6); (3) the use of a pretrained model - while TESS can still outperform other methods without using a pretrained model (Table 3), being able to make use of pretrained models with relatively little extra training.

Steps	QQP	NEWSELA-AUTO	QG	CNN-DM	MRPC	
	R-L				Accuracy	F1
10	<b>62.4</b>	58.4	38.8	35.6	<b>89.7</b>	<b>92.8</b>
100	62.0	59.1	<b>38.9</b>	39.6	<b>89.7</b>	<b>92.8</b>
1000	62.2	<b>59.4</b>	<b>38.9</b>	<b>39.7</b>	<b>89.7</b>	<b>92.8</b>

Table 7: Impact of number of sampling steps on performance. Our method achieves competitive results with as few as 10 or 100 steps on NLG tasks and 10 for MRPC.

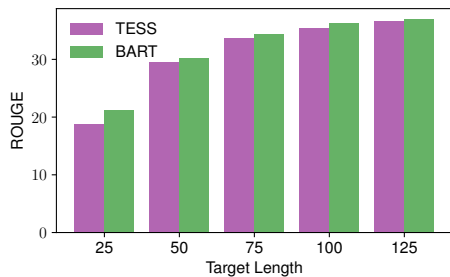


Figure 3: Average ROUGE score (R1, R2, R-L) for TESS vs BART for CNN-DM generations. Our method performs comparably to BART for different target lengths.

## 7 Related Work

**Diffusion for continuous domains** Diffusion models were first proposed by [Sohl-Dickstein et al. \(2015\)](#) and popularized by Denoising Diffusion Probabilistic Models (DDPMs) ([Ho et al., 2020](#)), which proposed a new parameterization that revealed an equivalence between ground-truth prediction and noise estimation. [Song et al. \(2021\)](#) proposed an alternative stochastic differential equation interpretation of diffusion that involves the Stein score function. [Nichol and Dhariwal \(2021\)](#) proposed a number of modifications to DDPMs, which improved log-likelihood and reduced sampling steps. [Ho and Salimans \(2021\)](#) proposed classifier-free guidance, which allows for highly controllable generation without the need for an external classifier to guide the model score estimates.

**Continuous diffusion for discrete domains** Following the success of diffusion models on continuous domains, there have been several attempts to apply diffusion on discrete data. [Li et al. \(2022a\)](#) applied diffusion on the latent token embedding space. Their resulting language model relies on word-level tokenization and works mostly on small datasets with a short sequence length of 64 tokens.

[Strudel et al. \(2022\)](#) used frozen pretrained word embedding with careful scaling to address the in-

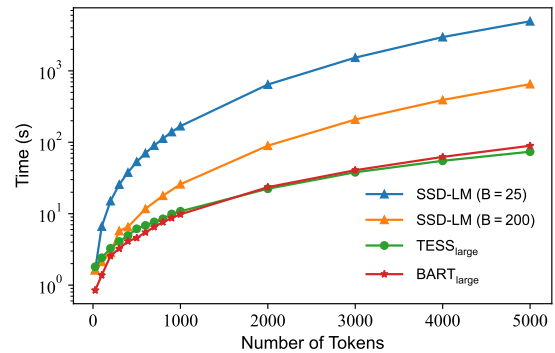


Figure 4: Time taken to generate variable number of tokens with 100 diffusion steps. We report the average time over five runs. Diffusion-based models use RoBERTa<sub>large</sub> as their backbone. TESS is substantially faster than SSD-LM. Notably, for 2000 tokens, it is even marginally faster than an equivalently-sized BART.

stability resulting from the competition between diffusion and reconstruction loss. However, their method does not allow the joint training of word embeddings, and the model was not evaluated on downstream NLP tasks. More recently, [Dieleman et al. \(2022\)](#) attempted to learn the embedding and diffusion model jointly, still by performing diffusion in the embedding space. Other recent works have also applied diffusion on word embeddings to tackle sequence-to-sequence problems ([Gong et al., 2023](#); [Yuan et al., 2022](#)). Concurrent to our work, [Ye et al. \(2023\)](#) proposed methods for manipulating the noise in the diffusion process during training and inference, yielding improved conditional text generation. Another concurrent work explores variational diffusion models for language modeling in embedding space ([Gulrajani and Hashimoto, 2023](#)). However, they compare their models to  $8\times$  smaller autoregressive models; our method obtains competitive performance with same-size autoregressive models.

Most relevant to our work, [Han et al. \(2022\)](#) proposed a semi-autoregressive diffusion model which generates small blocks of 25 tokens from



left to right, feeding them as additional context to generate next blocks. We extend their approach to fully non-autoregressive generation which substantially speeds up the inference time and incorporate an efficient self-conditioning method that exploits the semantics of the simplex space. Han et al. (2023b) similarly extends this approach, but focuses on showing the viability of simplex-based diffusion with large ( $> 1\text{B}$  parameter) models.

**Discrete diffusion for discrete domains** Unlike continuous diffusion models, discrete diffusion models maintain the discrete structure of the data domain and perform state transitions based on a probability matrix. Diffusion models with discrete state space were first explored by Sohl-Dickstein et al. (2015), who proposed a framework for diffusion over binary random variables. Later, Hoogeboom et al. (2021) and Austin et al. (2021) proposed discrete diffusion models for categorical random variables. However, these methods generally lag behind autoregressive models. More recently, Reid et al. (2022) proposed Diffuser, which formulates a discrete diffusion process by modeling generation as a series of discrete edit operations. TESS substantially outperforms Diffuser.

## 8 Conclusion

We present TESS, a new sequence-to-sequence diffusion model for language generation tasks that is fully non-autoregressive, works for long sequences compared to prior work, performs the diffusion process on the vocabulary logit space, and employs a new and efficient form of self-conditioning. TESS outperforms strong autoregressive baselines as well as recent state-of-the-art text diffusion models on a wide variety of conditional language generation and language understanding tasks, while also being far more efficient than prior diffusion-based models. Future work relies on pretraining our method combined with denoising and infilling objectives, which we hypothesize can provide further performance boosts to our text-to-text diffusion model.

## Limitations

**Sampling speed** As seen in Figure 4, TESS is still slower than BART when generating  $< 1000$  tokens. We experimented with reducing the number of diffusion steps (see Table 7), which can further speedup the generation. While in majority of tasks using just 10 steps provides promising results, it is not enough to achieve strong performance on more

complex tasks such as summarization. Incorporating recent work in computer vision to accelerate sampling in diffusion-based models (Song et al., 2023) could result in further speedups in generation.

**Long sequences** Inference speed tests with SSD-LM and BART show that TESS quickly dominates semi-autoregressive generation and outperforms BART at 2000 tokens. This result suggests that diffusion models have the potential of being faster than popular autoregressive models at long sequence lengths. In this work, we primarily used RoBERTa<sub>base</sub> models to facilitate fair comparison with existing baselines, which inevitably limited the size of the context window due to the absolute position embedding strategy employed by RoBERTa. We suspect that unlocking the full potential of diffusion-based language models may lie in the long sequence regime, which could involve scaling up the current models.

## Ethics Statement

Language models are known to produce toxic and biased content (Weidinger et al., 2022; Sheng et al., 2021). While we explore an alternate modelling framework to that commonly used in prior studies on the toxicity of language models, there is little reason to suggest our models would not also contain these issues. However, given the greater controllability of the diffusion framework (Li et al., 2022b), we hope future work explores how to make use of this controllability to reduce potential harms. Further examining how well results around toxic and harmful generations of autoregressive setups transfer to our setting may also aid in identifying future areas for improvement.

## Acknowledgements

We are grateful to Aman Madaan, Robin Strudel, Sander Dieleman, Chris Dyer, Xiaochuang Han, Sachin Kumar, Clara Meister, Sean Welleck, and Andre Wibisono for helpful comments and discussions, and Sam Skjonsberg and the ReViz team at AI2 for their support in managing experiments.

## References

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. 2021. Structured denoising diffusion models in discrete state-spaces. In *NeurIPS*.

- Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, and Danilo Giampiccolo. 2006. The second pascal recognizing textual entailment challenge. *Second PASCAL Challenges Workshop on Recognising Textual Entailment*.
- Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. 2009. The fifth pascal recognizing textual entailment challenge. In *TAC*.
- Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. **SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation**. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.
- Ting Chen, Ruixiang Zhang, and Geoffrey Hinton. 2022. Analog bits: Generating discrete data using diffusion models with self-conditioning. *arXiv preprint arXiv:2208.04202*.
- Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognizing textual entailment challenge. In *Machine Learning Challenges Workshop*.
- Aditya Deshpande, Jyoti Aneja, Liwei Wang, Alexander G. Schwing, and David A. Forsyth. 2018. Fast, diverse and accurate image captioning guided by part-of-speech. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10687–10696.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. **BERT: Pre-training of deep bidirectional transformers for language understanding**. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Bhuvan Dhingra, Kathryn Mazaitis, and William W Cohen. 2017. Quasar: Datasets for question answering by search and reading. *arXiv preprint arXiv:1707.03904*.
- Sander Dieleman, Laurent Sartran, Arman Roshanai, Nikolay Savinov, Yaroslav Ganin, Pierre H Richemond, Arnaud Doucet, Robin Strudel, Chris Dyer, Conor Durkan, et al. 2022. Continuous diffusion for categorical data. *arXiv preprint arXiv:2211.15089*.
- William B. Dolan and Chris Brockett. 2005. **Automatically constructing a corpus of sentential paraphrases**. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.
- 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*.
- Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. **The third PASCAL recognizing textual entailment challenge**. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9, Prague. Association for Computational Linguistics.
- Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and Lingpeng Kong. 2023. DiffuSeq: Sequence to sequence text generation with diffusion models. In *International Conference on Learning Representations, ICLR*.
- Jiatao Gu, Changhan Wang, and Junbo Zhao. 2019. Levenshtein transformer. *NeurIPS*.
- Ishaan Gulrajani and Tatsunori B Hashimoto. 2023. Likelihood-based diffusion language models. *arXiv preprint arXiv:2305.18619*.
- Xiaochuang Han, Sachin Kumar, and Yulia Tsvetkov. 2022. Ssd-lm: Semi-autoregressive simplex-based diffusion language model for text generation and modular control. *arXiv preprint arXiv:2210.17432*.
- Xiaochuang Han, Sachin Kumar, Yulia Tsvetkov, and Marjan Ghazvininejad. 2023a. **Ssd-2: Scaling and inference-time fusion of diffusion language models**. *ArXiv*, abs/2305.14771.
- Xiaochuang Han, Sachin Kumar, Yulia Tsvetkov, and Marjan Ghazvininejad. 2023b. **Ssd-2: Scaling and inference-time fusion of diffusion language models**.
- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *NeurIPS*.
- Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *NeurIPS*.
- Jonathan Ho and Tim Salimans. 2021. Classifier-free diffusion guidance. In *NeurIPS Workshop DGMs Applications*.
- Jonathan Ho, Tim Salimans, Alexey A Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. 2022. Video diffusion models. In *ICLR Workshop on Deep Generative Models for Highly Structured Data*.
- Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. 2021. Argmax flows and multinomial diffusion: Learning categorical distributions. *NeurIPS*.
- Chao Jiang, Mounica Maddela, Wuwei Lan, Yang Zhong, and Wei Xu. 2020. Neural crf model for sentence alignment in text simplification. In *ACL*.
- Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. 2020. Diffwave: A versatile diffusion model for audio synthesis. In *ICLR*.

- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*.
- Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander Rush, and Thomas Wolf. 2021. [Datasets: A community library for natural language processing](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori Hashimoto. 2022a. Diffusion-LM improves controllable text generation. In *NeurIPS*.
- Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B Hashimoto. 2022b. Diffusion-lm improves controllable text generation. In *NeurIPS*.
- Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Zhenghao Lin, Yeyun Gong, Yelong Shen, Tong Wu, Zhihao Fan, Chen Lin, Nan Duan, and Weizhu Chen. 2023. Text generation with diffusion language models: a pre-training approach with continuous paragraph denoise. In *Proceedings of the 40th International Conference on Machine Learning, ICML'23*. JMLR.org.
- 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.
- Alexander Quinn Nichol and Prafulla Dhariwal. 2021. Improved denoising diffusion probabilistic models. In *ICML*.
- 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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- 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. *NeurIPS*.
- 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*.
- 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. *JMLR*.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. [SQuAD: 100,000+ questions for machine comprehension of text](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.
- Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*.
- Machel Reid, Vincent J Hellendoorn, and Graham Neubig. 2022. Diffuser: Discrete diffusion via edit-based reconstruction. *arXiv preprint arXiv:2210.16886*.
- Pierre H Richemond, Sander Dieleman, and Arnaud Doucet. 2022. Categorical sdes with simplex diffusion. *arXiv preprint arXiv:2210.14784*.
- Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*.
- Nikolay Savinov, Junyoung Chung, Mikolaj Binkowski, Erich Elsen, and Aaron van den Oord. 2021. Step-unrolled denoising autoencoders for text generation. In *ICLR*.
- Kai Shen, Zeqian Ju, Xu Tan, Yanqing Liu, Yichong Leng, Lei He, Tao Qin, Sheng Zhao, and Jiang Bian. 2023. Naturalspeech 2: Latent diffusion models are natural and zero-shot speech and singing synthesizers. *ArXiv*, abs/2304.09116.
- Emily Sheng, Kai-Wei Chang, Prem Natarajan, and Nanyun Peng. 2021. [Societal biases in language generation: Progress and challenges](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International*

- Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4275–4293, Online. Association for Computational Linguistics.
- Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. [Recursive deep models for semantic compositionality over a sentiment treebank](#). In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.
- Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*.
- Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising diffusion implicit models. In *ICLR*.
- Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. 2023. Consistency models. In *ICML*.
- Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2021. Score-based generative modeling through stochastic differential equations. In *ICLR*.
- Robin Strudel, Corentin Tallec, Florent Althé, Yilun Du, Yaroslav Ganin, Arthur Mensch, Will Grathwohl, Nikolay Savinov, Sander Dieleman, Laurent Sifre, et al. 2022. Self-conditioned embedding diffusion for text generation. *arXiv preprint arXiv:2211.04236*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *NeurIPS*.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*.
- Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. [Neural network acceptability judgments](#). *Transactions of the Association for Computational Linguistics*, 7:625–641.
- Laura Weidinger, Jonathan Uesato, Maribeth Rauh, Conor Griffin, Po-Sen Huang, John Mellor, Amelia Glaese, Myra Cheng, Borja Balle, Atoosa Kasirzadeh, Courtney Biles, Sasha Brown, Zac Kenton, Will Hawkins, Tom Stepleton, Abeba Birhane, Lisa Anne Hendricks, Laura Rimell, William Isaac, Julia Haas, Sean Legassick, Geoffrey Irving, and Iason Gabriel. 2022. [Taxonomy of risks posed by language models](#). In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency, FAccT '22*, page 214–229, New York, NY, USA. Association for Computing Machinery.
- Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. [A broad-coverage challenge corpus for sentence understanding through inference](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *EMNLP: System Demonstrations*.
- Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. [Problems in current text simplification research: New data can help](#). *Transactions of the Association for Computational Linguistics*, 3:283–297.
- Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *TACL*.
- Jiasheng Ye, Zaixiang Zheng, Yu Bao, Lihua Qian, and Mingxuan Wang. 2023. [Dinoiser: Diffused conditional sequence learning by manipulating noises](#).
- Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Fei Huang, and Songfang Huang. 2022. Seqdiffuseq: Text diffusion with encoder-decoder transformers. *arXiv preprint arXiv:2212.10325*.
- Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2020. BERTscore: Evaluating text generation with BERT. In *ICLR*.



## A Experiment Details

### A.1 Dataset

NEWSLA-AUTO dataset (Jiang et al., 2020) is based on a Xu et al. (2015) with revised alignment and improved quantity and quality. For question generation, we use the QUASAR-T dataset (Dhingra et al., 2017); for summarization we use CNN-DailyMail (Hermann et al., 2015).

The GLUE benchmark (Wang et al., 2019) is released under the Creative Commons License (CC BY 4.0). This benchmark consists of multiple datasets: SST-2 (Socher et al., 2013), MNLI (Williams et al., 2018), CoLA (Warstadt et al., 2019), MRPC (Dolan and Brockett, 2005), QQP<sup>9</sup>, QNLI (Rajpurkar et al., 2016), STS-B (Cer et al., 2017), and RTE, which is a combination of data from RTE1 (Dagan et al., 2005), RTE2 (Bar-Haim et al., 2006), RTE3 (Giampiccolo et al., 2007), RTE5 (Bentivogli et al., 2009). We download all datasets from the Hugging Face Datasets library (Lhoest et al., 2021). Table 8 shows the sequence lengths for the source and target in each dataset, in number of tokens.

Dataset	Source	Target
GLUE	128	5
NEWSLA-AUTO	128	128
QQP	100	85
QG	155	65
CNN-DM	392	120

Table 8: Sequence length of each dataset.

### A.2 Baseline Details

When prior published results are on the same dataset and metric, or when the codebase is not publicly available, we use available reported results to reduce compute costs. To provide a fair comparison when tuning baselines ourselves (SSD-LM and BART), we use the same tuning budget and compared them in the same setting (see Appendix A.3). Lastly, we report values using the same decoding strategy, i.e., the default setting *without* minimum risk Bayes decoding (MBR).

For GENIE, we use the code and model weights provided by the authors<sup>10</sup> to evaluate GENIE using matched decoding settings to TESS.

<sup>9</sup><https://quoradata.quora.com/First-Quora-Dataset-Release-Question-Pairs>

<sup>10</sup><https://github.com/microsoft/ProphetNet/tree/master/GENIE>

### A.3 Training Hyperparameters

Following Han et al. (2022), we initialized the model from a pretrained model and similarly found that it improved performance (See Table 3). We implemented our work using HuggingFace Transformers (Wolf et al., 2020) and used the Huggingface Diffusers<sup>11</sup> to build our diffusion pipeline. Our experiments are performed on 8 NVIDIA A6000/A100 GPUs.

We trained our models and baselines with a learning rate of  $3e-5$  with the AdamW optimizer, with default parameters  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ ,  $\epsilon = 1e-8$ . We use a linear learning rate scheduler. We use the base model sizes for all experiments (Wolf et al., 2020).

For SSD-LM, we use a block size of 25 following the original paper and the same number of diffusion steps during training and inference as our own models. We reuse the codebase provided by the authors and adapt it to support downstream tasks. We do not test SSD-LM on CNN-DM due to the difficulty of training SSD-LM on outputs involving multiple decoding blocks, requiring custom data preprocessing, and further algorithm tweaks to handle the long outputs.

For all generation tasks, we train our method and baselines for paraphrase generation, summarization, question generation, and text simplification for 90K, 120K, 120K, and 80K steps, respectively. We set the number of warmup steps to 2000 for all generation tasks. For the experiments on GLUE, we set the number of warm-up steps to 500. We trained the models on larger datasets in GLUE for 25K steps; for smaller datasets, we use 12K steps. We then evaluate the models every 1K steps and report the results on the checkpoint obtaining the best results on the development set. We found that the training time of each model is roughly similar: with equivalent configurations on a single GPU on the QQP dataset, TESS achieves 1.7 train steps per second; SSD-LM, 1.8; BART, 1.4, using PyTorch 2.0.

### A.4 Evaluation Package Details

We use the following packages for calculating the given metric:

- BLEU: We use the sacrebleu package (Post, 2018), v2.3.1.

<sup>11</sup><https://huggingface.co/docs/diffusers>

- **ROUGE**: We use the rouge-score package, v0.2.1.<sup>12</sup>
- **Mauve**: We use the mauve-text package (Pillutla et al., 2021), v0.3.0.
- **BERTScore**: We use the bert-score package (Zhang et al., 2020), v0.3.12.

For other metrics, we use our own implementations (usually heavily based on a reference implementation), which will be open-sourced.

### A.5 Inference Speed Experiments

For the inference speed numbers reported in Table 4, we run all experiments on a single 80 GB NVIDIA A100 GPU. We use an adapted version of the SSD-LM inference script provided by the authors in their public repository, removing logging and initializing tensors on-device to avoid expensive `.to()` calls. For BART<sub>large</sub>, we use the transformers library (Wolf et al., 2020). We alter all models to allow sequence lengths over 512 tokens by resizing the position embeddings matrix. We use the following context: “A man of innumerable personalities and powers vs. the most powerful artificial intelligence in this universe: Legion vs. Nimrod! With Nightcrawler in Orchis clutches, David Haller and his allies will have to confront the mastermind who”. We report the exact timings and standard deviations in Table 10.

## B Sampling Steps

We performed additional ablations on the relationship between the number of sampling steps and performance on question generation.

Model	Steps	Forwards	R-L
BART	—	74	38.8
SSD-LM	10	50	33.0
SSD-LM	100	500	36.7
SSD-LM	1000	5000	38.5
TESS	10	10	38.8
TESS	100	100	<b>38.9</b>
TESS	1000	1000	<b>38.9</b>

Table 9: Sampling step ablation on question generation.

We note forward passes are not directly comparable between AR and NAR models: AR models may use more or less forwards than NAR models depending on the number of tokens generated, and

<sup>12</sup><https://github.com/google-research/google-research/tree/master/rouge>

each forward pass of the AR model involves a differing number of tokens. Here, we use 74, as this is the average number of BART tokens in question generation responses.

Overall, we observe that SSD-LM’s performance drops significantly with fewer diffusion steps, while TESS remains largely unaffected. Notably, TESS achieves parity with BART with only 10 sampling steps.

## C Inference Step

In the typical variant of DDPM, the model predicts the added noise  $\epsilon_{\theta}(\mathbf{x}_t, t)$  instead of original signal and the DDPM inference step (Ho et al., 2020) is as follows:<sup>13</sup>

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left( \mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{1 - \alpha_t}} \epsilon_{\theta}(\mathbf{x}_t, t) \right) \quad (13)$$

Since we work with the variant predicting the signal itself, we substitute (2) into (13), obtaining:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left( \sqrt{\bar{\alpha}_t} \mathbf{x}_0 - \frac{\alpha_t - \bar{\alpha}_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_{\theta}(\mathbf{x}_t, t) \right). \quad (14)$$

Given that  $\bar{\alpha}_t = \bar{\alpha}_{t-1} \alpha_t$ , we arrive at:

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0 - \frac{\sqrt{\alpha_t - \bar{\alpha}_t} \sqrt{\alpha_t - \bar{\alpha}_t}}{\sqrt{\alpha_t} \sqrt{1 - \bar{\alpha}_t}} \epsilon_{\theta}(\mathbf{x}_t, t). \quad (15)$$

Using a cosine schedule for  $\bar{\alpha}_t$  (Nichol and Dhariwal, 2021),  $\sqrt{(\alpha_t - \bar{\alpha}_t)/(1 - \bar{\alpha}_t)} \geq 0.98$  for 98%  $t \in (1, T)$ , with some outliers as  $t \rightarrow 0$  and  $t \rightarrow T$  (Han et al., 2022). Thus, with the approximation  $\sqrt{(\alpha_t - \bar{\alpha}_t)/(1 - \bar{\alpha}_t)} \approx 1$ , Equation (15) further simplifies into:

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0 - \sqrt{1 - \bar{\alpha}_{t-1}} \epsilon_{\theta}(\mathbf{x}_t, t). \quad (16)$$

In our case, the signal  $\mathbf{x}_t$  is the simplex  $\mathbf{S}_t$ , with  $\hat{\mathbf{S}}_{\theta}$  as the model prediction of the ground-truth. Adjusting the above with this notation, we recover Equation (9):

$$\mathbf{S}_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \hat{\mathbf{S}}_{\theta}(\mathbf{S}_t, t) + \sqrt{1 - \bar{\alpha}_{t-1}} \epsilon_t.$$

## D Qualitative Examples

We show randomly chosen example outputs from TESS model and BART, the strongest baseline, on the summarization task in Table 11. Qualitatively, we observe that TESS is capable of generating natural samples that are often indistinguishable from those of BART.

<sup>13</sup>Following Han et al. (2022) we drop the additional noise term  $\sigma_t \mathbf{z}$ , where  $\mathbf{z} \in \mathcal{N}(0, \mathbf{I})$ .

Model	Blocks	Number of Tokens														
		25	100	200	300	400	500	600	700	800	900	1000	2000	3000	4000	5000
SSD-LM	25	1.6 <sub>0.3</sub>	6.6 <sub>0.3</sub>	15.0 <sub>0.3</sub>	25.6 <sub>0.3</sub>	37.8 <sub>0.3</sub>	53.4 <sub>0.4</sub>	70.4 <sub>0.0</sub>	90.1 <sub>0.0</sub>	112.7 <sub>0.0</sub>	139.4 <sub>0.0</sub>	168.4 <sub>0.0</sub>	643.2 <sub>0.2</sub>	1531.5 <sub>0.6</sub>	2933.4 <sub>0.9</sub>	4945.1 <sub>2.7</sub>
SSD-LM	200	1.6 <sub>0.3</sub>	2.1 <sub>0.3</sub>	2.8 <sub>0.3</sub>	5.8 <sub>0.3</sub>	6.5 <sub>0.3</sub>	-	11.7 <sub>0.3</sub>	-	17.9 <sub>0.0</sub>	-	25.8 <sub>0.0</sub>	89.6 <sub>0.0</sub>	207.8 <sub>0.0</sub>	391.1 <sub>0.2</sub>	652.2 <sub>0.2</sub>
BART <sub>large</sub>	-	0.8 <sub>0.2</sub>	1.4 <sub>0.2</sub>	2.6 <sub>0.2</sub>	3.2 <sub>0.2</sub>	4.1 <sub>0.2</sub>	4.6 <sub>0.2</sub>	5.5 <sub>0.0</sub>	6.5 <sub>0.0</sub>	7.6 <sub>0.0</sub>	8.7 <sub>0.0</sub>	9.8 <sub>0.0</sub>	23.5 <sub>0.2</sub>	40.6 <sub>0.1</sub>	62.4 <sub>0.1</sub>	89.4 <sub>0.4</sub>
TESS	-	1.8 <sub>0.3</sub>	2.4 <sub>0.3</sub>	3.3 <sub>0.3</sub>	4.1 <sub>0.3</sub>	4.9 <sub>0.3</sub>	6.1 <sub>0.3</sub>	6.9 <sub>0.0</sub>	7.7 <sub>0.0</sub>	8.5 <sub>0.0</sub>	9.9 <sub>0.0</sub>	10.8 <sub>0.0</sub>	22.3 <sub>0.0</sub>	38.0 <sub>0.0</sub>	55.0 <sub>0.0</sub>	73.8 <sub>0.0</sub>

Table 10: Time taken to generate the given number of tokens with a 50-token prefix in seconds. All models use 100 diffusion steps and RoBERTa<sub>large</sub> as the underlying model. All values are the average over 5 runs, with standard deviations given as subscripts; standard deviations less than 0.05 appear as 0.0.

Gold	BART	Ours
Membership gives the ICC jurisdiction over alleged crimes committed in Palestinian territories since last June.	The Palestinian Authority officially becomes the 123rd member of the International Criminal Court.	The Palestinian Authority is the 123rd member of the International Criminal Court.
Israel and the United States opposed the move, which could open the door to war crimes investigations against Israelis.	It gives the court jurisdiction over alleged crimes committed in Palestinian territories.	The move gives it jurisdiction over alleged crimes in Palestinian territories.
	Israel and the United States, neither of which is an ICC member, opposed the Palestinians' efforts to join the body.	Israel and the United States oppose the Palestinians' efforts to join it.
Theia, a bully breed mix, was apparently hit by a car, whacked with a hammer and buried in a field.	A stray pooch in Washington State has used up at least three of her own injuries.	A stray dog was found on a farm in Washington State.
"She's a true miracle dog and she deserves a good life," says Sara Mellado, who is looking for a home for Theia.	Theia, a bully breed mix, was hit by a car and then buried in a field.	The dog has used up at least three of her own after being hit in the head by a car and being buried in a field.
	She has been receiving care at the Veterinary Teaching Hospital.	Theia has a dislocated jaw, leg injuries and needs surgery to help breathe.
Mohammad Javad Zarif has spent more time with John Kerry than any other foreign minister.	Mohammad Javad Zarif is the Iranian foreign minister.	Mohammad Javad Zarif is now Iran's foreign minister.
He once participated in a takeover of the Iranian Consulate in San Francisco.	He has been John Kerry's opposite number in securing a breakthrough in nuclear talks.	He has been John Kerry's opposite number in securing a breakthrough in nuclear talks.
The Iranian foreign minister tweets in English.		
17 Americans were exposed to the Ebola virus while in Sierra Leone in March.		Five Americans were monitored for three weeks at an Omaha hospital.
Another person was diagnosed with the disease and taken to hospital in Maryland.	One of the five had a heart-related issue on Saturday and has been discharged.	One of the five had a heart-related issue on Saturday.
National Institutes of Health says the patient is in fair condition after weeks of treatment.	The others have already gone home.	The others have already gone home.
		They were exposed to Ebola in Sierra Leone, but none developed the deadly virus.
Student is no longer on Duke University campus and will face disciplinary review.	A student admitted to hanging a noose made of rope from a tree near a student union.	A Duke student admitted to hanging a noose from a tree.
School officials identified student during investigation and the person admitted to hanging the noose, Duke says.	Duke didn't identify the student, citing federal privacy laws.	The private school didn't identify the student, citing federal privacy laws.
The noose, made of rope, was discovered on campus about 2 a.m.	The incident is one of several recent racist events to affect college students.	The student is no longer on campus and will face student conduct review.
College-bound basketball star asks girl with Down syndrome to high school prom.	Eastern High School basketball player Trey Moses asked his girlfriend to be his prom date.	College basketball player and high school freshman picked Ellie Meredith as his prom date.
Pictures of the two during the "prom-posal" have gone viral.	Ellie Meredith, a freshman with Down syndrome, has struggled with friendships since elementary school.	The photos have gone viral on social media.
	A special program at Eastern has made things easier for her, her mom says.	
Amnesty's annual death penalty report catalogs encouraging signs, but setbacks in numbers of those sentenced to death.	Amnesty International says governments are using the death penalty to advance executions.	Amnesty International released its annual report on the death penalty.
Organization claims that governments around the world are using the threat of terrorism to advance executions.	At least 607 people were executed around the world in 2014, compared to 778 in 2013.	At least 607 people were executed in 2014, compared to 778 in 2013.
The number of executions worldwide has gone down by almost 22% compared with 2013, but death sentences up by 28%.		Report cites Pakistan lifting a six-year moratorium on the execution of civilians.
		China has used the death penalty as a tool in its "Strike Hard" campaign against terrorism in the restive far-western province of Xinjiang.
Andrew Getty's death appears to be from natural causes, police say, citing coroner's early assessment.	Andrew Getty appears to have died of natural causes, police say.	Coroner's preliminary assessment says there was no foul play involved in the death of Getty.
In a petition for a restraining order, Getty had written he had a serious medical condition.	The coroner's preliminary assessment is there was no foul play involved in his death.	Andrew Getty, 47, had "several health issues," detective says.
Police say this is not a criminal matter at this time.	He was the grandson of oil tycoon J. Paul Getty.	There is no criminal investigation underway, detective says.
Once a super typhoon, Maysak is now a tropical storm with 70 mph winds.	Maysak has lost a lot of steam as it spins west in the Pacific Ocean.	Maysak gained super typhoon status thanks to sustained 150 mph winds.
It could still cause flooding, landslides and other problems in the Philippines.	It boasts steady winds of more than 70 mph (115 kph) and gusts up to 90 mph.	Authorities have forbidden outdoor activities like swimming, surfing, diving and boating in some locales.

Table 11: Randomly chosen samples generated on the CNN-DM dataset by BART and TESS.