CS6420 Topics in Deep Learning
Fine Grained Text Style Transfer - Final Report

Anudeep Rao Perala (CS21BTECH11043)
Asli Nitej Reddy Busireddy (CS21BTECH11011)

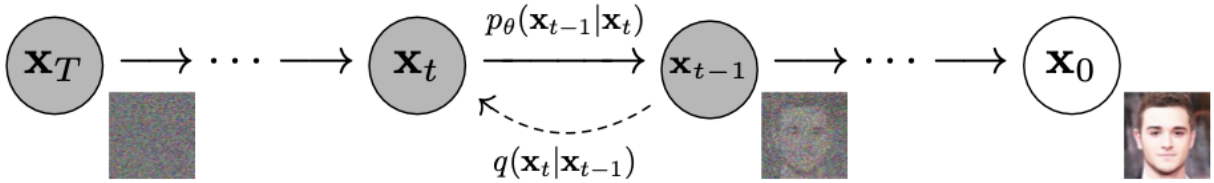
Contents

1 Problem Statement

We target the Seq2Seq text generation task for fine grained control over text-to-text modification. Given a m-length source sequence $w^x = w_1^x, ..., w_m^x$ and style tokens $s = s_1, , ..., s_k$ we aim to learn a diffusion model that can produce a n-length target sequence $w_y = w_1^y, ..., w_n^y$ conditioning on the source sequence and the style tokens.

2 Background of the Problem

2.1 Introduction to Diffusion



- The process of diffusion consists of two processes (i) **Forward Process** and (ii) **Backward Process**.

2.1.1 Forward Process

- The process begins with a real data point x_0.
- During the forward process, the model gradually adds random Gaussian noise to the data in a series of steps (from 1 to T). At each step t, the noise is carefully controlled by a mathematical function $q(x_t|x_{t-1})$, which follows a Gaussian distribution with specific parameters.
- The noise addition at each step is governed by the equation:

$$q(x_t \mid x_{t-1}) = \mathcal{N}\left(x_t;\ \sqrt{1-\beta_t}x_{t-1}, \beta_t I\right) \tag{1}$$

- β_t is a schedule parameter that controls how much noise is added at each step $\beta_t \in (0, 1)$.
- This process continues until the data becomes pure Gaussian noise
- Once the forward process is completed, the reverse denoising process tries to gradually reconstruct the original data x_0 via sampling from x_T by learning a diffusion model f_θ.

2.1.2 Backward Process

- After the forward process transforms the original data into Gaussian noise (x_t), the model learns to reverse this corruption. This is where the real power of diffusion models lies. The reverse process, parameterized by a learned model f_θ, attempts to gradually remove the noise and reconstruct the original data distribution.
- The process can be visualized as a Markov chain, where each state depends only on the previous state, both in the forward and reverse directions.
- This structured approach to adding and removing noise makes diffusion models to capture the noise to be removed from random noise to generate the data space.

2.2 Math in Diffusion

- After doing the required math we end up at the following equation:

$$-\log p_\theta(x_0) \leq \mathbb{E}_{q(x_{1:T}|x_0)}\left[-\log \frac{p(x_{0:T})}{q(x_{1:T}|x_0)}\right] \tag{2}$$

2.2.1 The ELBO Term

The right-hand side contains what's known as the Evidence Lower BOund (ELBO). This term is expressed as an expectation (E) over the forward process distribution $q(x_{1:T}|x_0)$ and calculating the $\left[-\log \frac{p(x_{0:T})}{q(x_{1:T}|x_0)}\right]$.

2.2.2 About the Inequality

- The inequality states that the negative log likelihood we want to minimize is upper-bounded by the ELBO term. This is particularly useful because while directly minimizing $-\log(p_\theta(x_0))$ is intractable, we can instead work on minimizing the ELBO.
- By minimizing the ELBO we can achieve the following:
  * Trains our model to better understand the relationship between noisy and clean data.
  * Improves our model's ability to reverse the diffusion process which achieves better generation quality when sampling from the model.

2.2.3 Math contd.

- Upon expanding ELBO we get the final expression:

$$= -\underbrace{\mathbb{E}_{q(x_1|x_0)}\left[\log p_\theta(x_0|x_1)\right]}_{L_0} + \underbrace{D_{KL}(q(x_T|x_0)\|p(x_T))}_{L_T} + \underbrace{\sum_{t=2}^{T} \mathbb{E}_{q(x_t|x_0)}\left[D_{KL}(q(x_{t-1}|x_t, x_0)\|p_\theta(x_{t-1}|x_t))\right]}_{L_{t-1}} \tag{3}$$

- Meaning of the terms:
  * L_0**:** This can be interpreted as the reconstruction term.
  * L_T**:** This term has no optimization as it has no parameters and for large T the final distribution is Gaussian which makes this term zero.
  * L_t**:** Tries to make the distribution at x_t consistent, from both forward and backward processes. We try to minimize this.
- We want to minimize the following term:

$$\arg\min_\theta D_{KL}(q(x_{t-1}|x_t, x_0)\|p_\theta(x_{t-1}|x_t)) \tag{4}$$

- We set the variances of the two Gaussian's to match exactly, upon performing the mathematical steps we end at the result where, optimizing the KL Divergence term reduces to minimizing the difference between the means of the two distributions.

$$\arg\min_\theta \frac{1}{2\sigma_q^2(t)} \left\|\mu_\theta(x_t, t) - \mu_q(x_t, x_0)\right\|_2^2 \tag{5}$$

- Upon simplifying the equation:

$$q(x_{t-1}|x_t, x_0) = \frac{q(x_t|x_{t-1}, x_0)q(x_{t-1}|x_0)}{q(x_t|x_0)} \tag{6}$$

- We end up at a relation:

$$q(x_{t-1}|x_t, x_0) \propto \mathcal{N}\left(x_{t-1}; \underbrace{\frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})x_t + \sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)x_0}{1-\bar{\alpha}_t}}_{\mu_q(x_t,x_0)}, \underbrace{\frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{I}}_{\Sigma_q(t)}\right) \tag{7}$$

- So from the above equation we get the value of $\mu_q(x_t, x_0)$.

- We can now similarly formulate $\mu_\theta(x_t, t)$ by setting it to the following form:

$$\mu_\theta(x_t, t) = \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})x_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)\hat{x}_\theta(x_t, t)}{1 - \bar{\alpha}_t} \tag{8}$$

- Where $\hat{x}_\theta(x_t, t)$ is parameterized by a neural network that predicts x_0 from noisy image x_t and time index t.
- After substituting the above values of $\mu_q(x_t, x_0)$ and $\mu_\theta(x_t, t)$ in eq(5) the optimization problem simplifies to:

$$\arg\min_\theta \frac{1}{2\sigma_q^2(t)} \frac{\bar{\alpha}_{t-1}(1 - \alpha_t)^2}{(1 - \bar{\alpha}_t)^2} \|\hat{x}_\theta(x_t, t) - x_0\|_2^2 \tag{9}$$

- Now we will be using the eq(9) as our loss function in while training the diffusion model.

2.3 Text Style Transfer

- **Text style transfer** represents a sophisticated natural language processing task that aims to transform text while maintaining a delicate balance between two crucial aspects: preserving the fundamental meaning of the original text while deliberately modifying its stylistic characteristics.
- **Fine grained text style transfer** takes this concept further by introducing precise control over specific stylistic elements. Unlike broader approaches that might simply transform text from formal to informal, fine grained TST allows for more control over modifying the aspects of the text while carefully preserving the original meaning.
- This fine grained style transfer can be categorized into 4 categories for transfer:
  - Lexical transfer
  - Syntax transfer
  - Semantic transfer
  - Thematic transfer

2.3.1 Lexical transfer

Involves word level changes focusing on vocabulary and word meanings. operations such as replacing words with their synonyms or antonyms.
ex: The cat is very quick $\xrightarrow{\text{synonym change}}$ The cat is very slow

2.3.2 Syntax transfer

Syntax Transfer: Modifies grammatical structures without altering the content. Involves transforming sentence elements like tense, voice, or proposition positions.
ex: She writes a letter $\xrightarrow{\text{past tense}}$ She wrote a letter

2.3.3 Semantic transfer

These changes affect the meaning of the sentence which includes removing or adding information. These changes are beyond just word or syntax-level modifications.
ex: The dog is barking loudly at the stranger $\xrightarrow{\text{Info removal}}$ The dog is barking at the stranger (adj. loudly is removed)

2.3.4 Thematic transfer

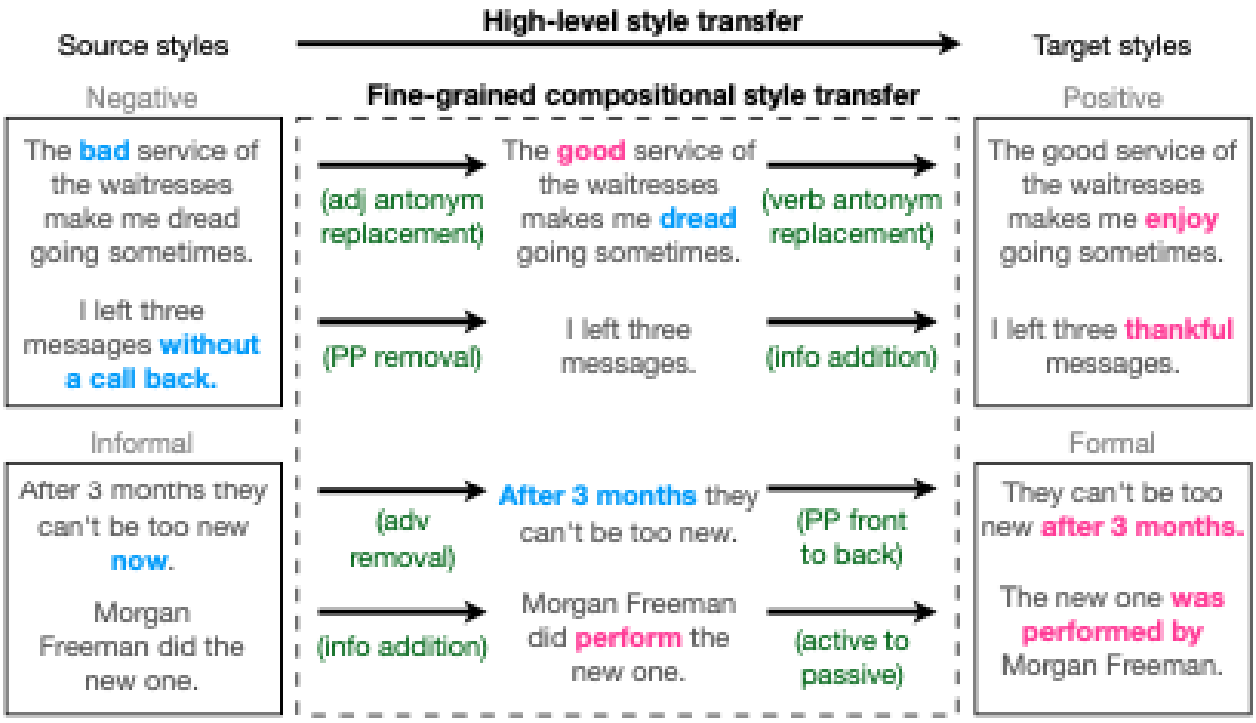
Adjusts the emphasis within a sentence to highlight different parts to shift the perspective or importance.
ex: The comparable year-earlier number was 56 million a spokesman said $\xrightarrow{\text{attention}}$ A spokesman said the year-earlier number of 56 million was comparable (56 million $\rightarrow$ comparable)

2.4 About the Dataset

- We define the fine-grained style constructs over 21 categories.
- The 21 categories are as follows:

Aspect	Transfer	Original Sentence	Additional Info/ Emphasis	Transferred Sentence
LEXICAL	Noun synonym replacement	The **shift** wo n't affect operations.		The **displacement** wo n't affect operations.
	Noun antonym replacement	Investors will develop thicker skins and their **confidence** will return he says.		Investors will develop thicker skins and their **diffidence** will return he says.
	Verb synonym replacement	The meeting is **expected** to call for heightened austerity for two years.		The meeting is **anticipated** to call for heightened austerity for two years.
	Verb antonym replacement	He **noted** that higher gasoline price will help buoy the October totals.		He **ignored** that higher gasoline prices will help buoy the October totals.
	ADJ synonym replacement	Most other states have enacted **similar** bans.		Most other states have enacted **alike** bans.
	ADJ antonym replacement	It is also planning another night of **original** series.		It is also planning another night of **unoriginal** series.
	Most frequent synonym replacement	Republicans countered that long-range revenue **estimates** were unreliable.		Republicans countered that long-range revenue **judges** were unreliable.
	Least frequent synonym replacement	Merrill Lynch Capital Markets Inc. is the sole **underwriter** for the **offering** .		Merrill Lynch Capital Markets Inc. is the sole **investment-banker** for the **oblation** .
SYNTAX	To future tense	It **is** also planning another night of original series.		It **will be** also planning another night of original series.
	To present tense	Sen. Mitchell **urged** them to desist.		Sen. Mitchell **urges** them to desist.
	To past tense	It **is** also planning another night of original series.		It **was** also planning another night of original series.
	Active to passive	He also received 20-year sentences for each of the 24 passengers injured.		20-year sentences also were received by him for each of the 24 passengers injured.
	Passive to active	Most bills are drafted by bureaucrats not politicians.		Bureaucrats not politicians draft most bills.
	PP front to back	**In Indianapolis** Lilly declined comment.		Lilly declined comment **in Indianapolis** .
	PP back to front	The dollar has been strong **unlike 1987** .		**Unlike 1987** the dollar has been strong.
SEMANTICS	ADJ or ADV removal	The controls on cooperatives appeared **relatively** liberal when first introduced		The controls on cooperatives appeared liberal when introduced
	PP removal	The controls **on cooperatives** appeared relatively liberal when first introduced.		The controls appeared relatively liberal when first introduced.
	Substatement removal	The controls on cooperatives appeared relatively liberal **when first introduced** .		The controls on cooperatives appeared relatively liberal.
	Information addition	He reports his business is up slightly from customers replacing old stock.	[**'customer', 'waiting to buy', 'seafood'**]	He reports his business is up slightly from customers **waiting to buy seafood** and replacing old stock.
THEMATICS	Verb/Action emphasis	He intends to add to the litigation staff.	**add**	**Adding** to the litigation staff is what he intends to do.
	Adjective emphasis	The comparable year-earlier number was 56 million a spokesman said.	**comparable**	A spokesman said the year-earlier number of 56 million **was comparable** .

- The **Relative Difficulty of Transfers** is calculated at the token- level (i.e. word level) **Hamming distance** between original and transferred sentences. Using this metric we categorized these 13 transfers into **easy**, **medium** and **hard** categories.
- The following example emphasises the process of using fine-grained style constructs to achieve a high level style transfer.



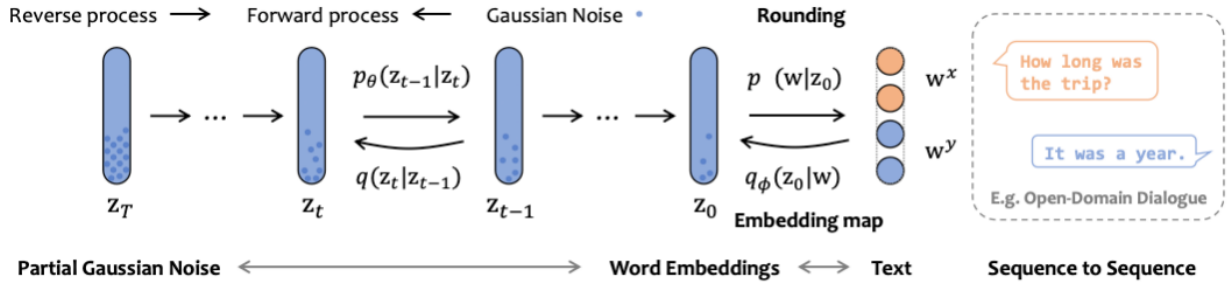
3 Previous Work on Seq2Seq Diffusion

Now we look into the setting of diffusion where the target sequence is on a source sequence. Extending the continuous diffusion models to natural language is a challenge because of the discrete nature of texts. So in order to address this we use an embedding function and a rounding function. The diffusion is called **DiffuSeq**.

3.1 Problem Statement

We target the sequence-to-sequence text generation tasks. In particular, given a m-length source sequence $w_x = w_x^1, ..., w_x^m$, we aim to learn a diffusion model that can produce a n-length target sequence $w_y = w_y^1, ..., w_y^n$ conditioning on the source sequence.

3.2 Seq2Seq Diffusion



In this diffusion also we have the two standard process (i) **Forward Process** and (ii) **Backward Process**

3.2.1 Forward Process with Partial Noising

- We have an embedding function EMB(w). given a pair of source sequence w^x and target sequence w^y DiffuSeq tries to learn the unified feature space of $\text{EMB}(\mathbf{w}^{x\oplus y}) \in \mathbb{R}^{(m+n)\times d}$. ($|w^x| = m, |w^y| = n$). The $\text{EMB}(\mathbf{w}^{x\oplus y})$ is defined as, $\text{EMB}(\mathbf{w}^{x\oplus y}) = [Emb(w_1^x), ..., EMB(w_m^x), Emb(w_1^y, ..., EMB(w_n^y))]$. We model this sequence of embedding to a new markov transition parametrized $q_\phi(z_0|\mathbf{w}^{x\oplus y}) = \mathcal{N}(\text{EMB}(\mathbf{w}^{x\oplus y}), \beta_0\mathbf{I})$.

- We now define $z_t = x_t \oplus y_t$, where $x_t \in w^x$ and $y_t \in w^y$. In each forward step $q(\mathbf{z}_t|\mathbf{z}_{t-1})$, gradually inject noise into last step's hidden state z_{t-1} to obtain z_t. Unlike conventional diffusion models that corrupt the whole z_t (both x_t and y_t) without distinction, we only impose noising on y_t. This modification is called **Partial Noising** which allows us to adapt diffusion models for **conditional language modeling**.

3.2.2 Reverse Process

- Our goal is to recover original z_0 by denoising the z_t. By the learning process $p_\theta(\mathbf{z}_{t-1}|\mathbf{z}_t) = \mathcal{N}(\mathbf{z}_{t-1}; \mu_\theta(\mathbf{z}_t, t), \sigma_\theta(\mathbf{z}_t, t))$. μ_θ and σ_θ are parameters for the predicted value of z_0.

- We model $f_\theta(z_t, t)$ as our neural network, we use transformer architecture to model f_θ, which models the semantic relation between x_t and y_t. The model tries to learn the diffusion models neural network and the embedding layers parameters jointly.

- The variational lower bound(L_{vlb}) is formulated as:

$$\mathcal{L}_{\text{vlb}}(\mathbf{w}) = \mathbb{E}_{q_\phi(z_0|\mathbf{w})}\left[\mathcal{L}_{\text{vlb}}(z_0) + \log q_\phi(z_0|\mathbf{w}) - \log p(\mathbf{w}|z_0)\right] \tag{10}$$

  Where $L_{\text{vlb}}(z_0)$ corresponds to the standard variational lower bound in diffusion.

- On modifying eq(10) more we get:

$$\mathcal{L}_{\text{vlb}}(\mathbf{w}) = \mathbb{E}_{q_\phi(z_{0:T}|\mathbf{w})}\Bigg[\underbrace{\log\frac{q(z_T|z_0)}{p_\theta(z_T)}}_{L_T} + \sum_{t=2}^{T}\underbrace{\log\frac{q(z_{t-1}|z_0,z_t)}{p_\theta(z_{t-1}|z_t)}}_{L_{t-1}} + \underbrace{\log\frac{q_\phi(z_0|\mathbf{w})}{p_\theta(z_0|z_1)}}_{L_0} - \underbrace{\log p(\mathbf{w}|z_0)}_{L_{\text{round}}}\Bigg] \tag{11}$$

- After applying all the approximations on eq(11) like how we do in standard diffusion models we get:

$$\mathcal{L}_{\text{vlb}}(\mathbf{w}) = \min_\theta\left[||\mu(z_T)||^2 + \sum_{t=2}^{T}||z_0 - f_\theta(z_t, t)||^2 \quad + ||\text{EMB}(\mathbf{w}^{x\oplus y}) - f_\theta(z_1, 1)||^2 + \mathcal{R}(||\mathbf{z}_0||^2)\right] \tag{12}$$

- During training the model estimates the z_0 via $f_\theta(z_t, t)$. One point to note is that although in the first term, we only compute the loss w.r.t y_0, due to the attention mechanism in the transformer, the reconstruction of y_0 also takes x_0 into account, thus the gradients from the first term will also affect the learning of x_0.

- The term $\mathcal{R}(||\mathbf{z}_0||^2)$ is introduced to learn regularize the embedding learning. We share the embedding function between source and target sequences, enabling the training of two different feature spaces jointly.

3.2.3 Inference

- Given the condition $\text{EMB}(w^x)$, we randomly sample $y_T \sim N(0, I)$ and concatenate y_T with $\text{EMB}(w^x)$ to obtain z_T. We now repeat the reverse process until we arrive at z_0 by calculating z_0^{temp}. Now using this z_0^{temp}, we sample $\mathbf{z}_{t-1}$ from $q(\mathbf{z}_{t-1} \mid f_\theta(\mathbf{z}_t, t), \mathbf{z}_t)$, which is fed as input to the next diffusion step.

- The equation for obtaining $\mathbf{z}_{t-1}$:

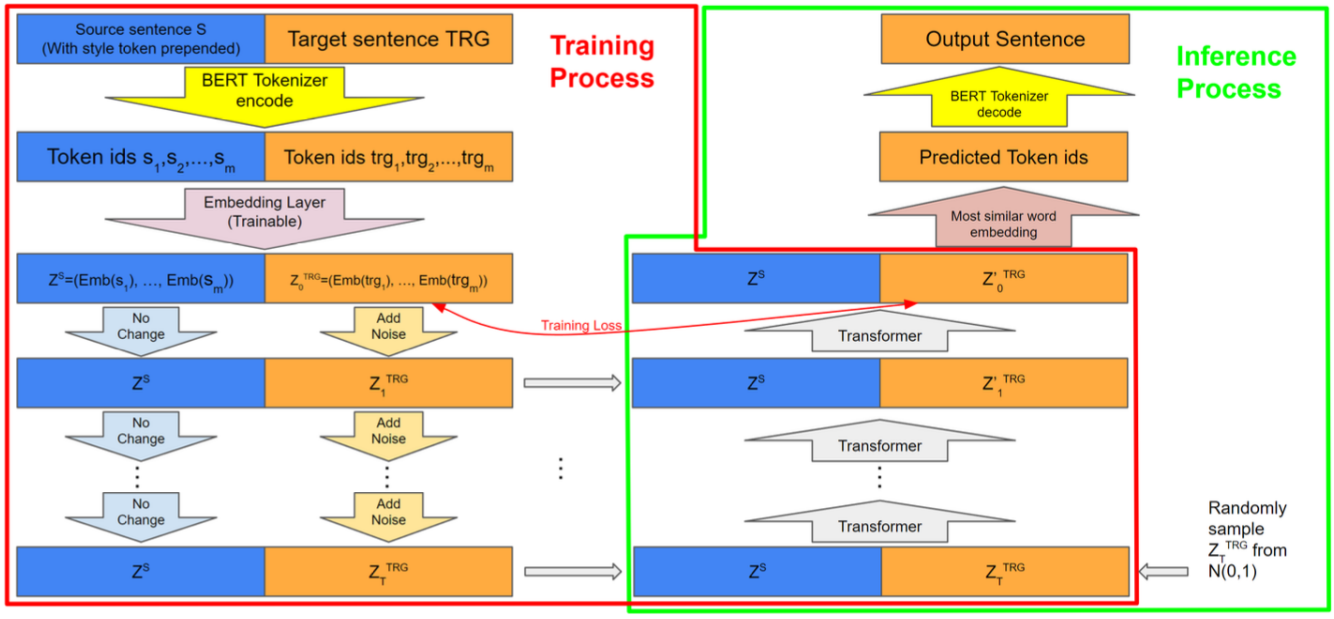
$$\mathbf{z}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}f_\theta(\mathbf{z}_t, t) + \sqrt{1-\bar{\alpha}_{t-1}}\epsilon, \text{where } \bar{\alpha}_t = \prod_{i=0}^{t}(1-\beta_i) \tag{13}$$

- At each sampling step anchoring function is executed towards the obtained z_{t-1} which does:
  - Rounds the obtained z_{t-1} back to word embedding space.
  - Replaces the part of recovered z_{t-1} that belongs to w_x with the original x_0. This replacement is essential in-order to maintain the condition on the input sentence for generating the required fine grained style transferred text.
- To improve the quality of generation, we apply the widely used **Minimum Bayes Risk (MBR)** decoding strategy. We first generate a set of candidate samples S from different random seeds of DIFFUSEQ and select the best output sequence that achieves the minimum expected risk under a meaningful loss function (We use BLEU as the metric).

4 SOTA for Fine Grained Text Style transfer

Adopted **DiffuSeq** for performing fine-grained text style transfer. We first define a set of special style tokens, one for each possible individual fine- grained transfer. If we wish to perform one or more transfer on the source sentence, we will prepend the corresponding special token(s) to the beginning of the source sentence to form the condition S. This process of defining the set of special tokens is the main step in this approach.

4.1 Architecture



4.2 Training

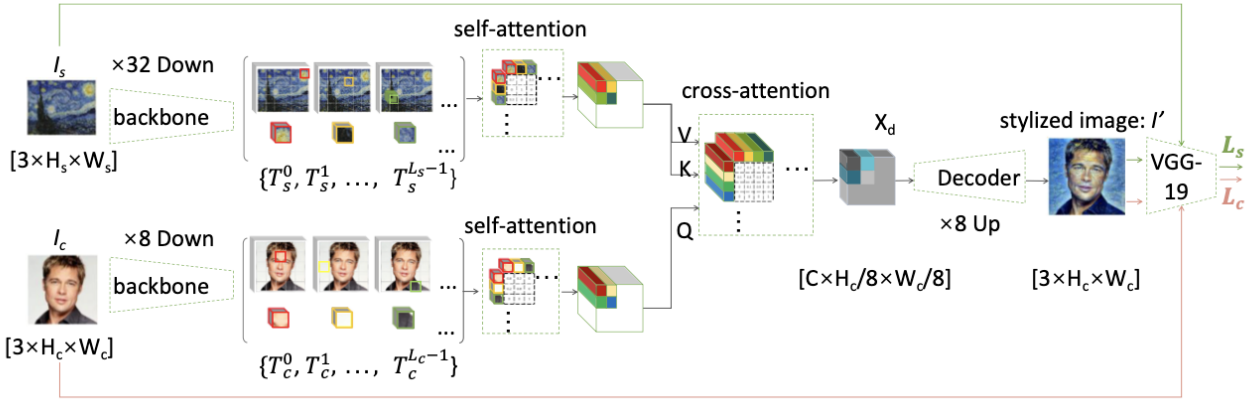
- Used **BERT** tokenizer for converting text into tokens. Then we include a **token embedding layer** to encode both the source(**prepended** with style tokens) and target.
- Both the diffusion transformer and the token embeddings are initialized randomly and jointly optimized.
- Z^S are source embeddings and Z_0^{TRG} are target embeddings. We then apply the partial noise in forward process until $t \sim U(1,T)$, after which we get Z_t^{TRG}. We then concatenate Z^S and Z_t^{TRG} input that to the diffusion transformer. Uses eq(12) as the loss for minimization.

4.3 Inference

- We randomly initialize $Z_T^{*TRG} \sim N(0,1)$, and encode the condition (source sentence and style tokens) into Z^S. Then we concatenate them and use the transformer to predict a temporary $Z_{0_{temp}}^{*TRG}$, then we add $\mathbf{T-1}$ steps of noise to obtain Z_{T-1}^{*TRG}. Now for each embedding in finally obtained Z_0^{TRG}, we find the closest embedding in our token embedding layer by cosine distance, and decode the embedding to that token.
- Then we combine the tokens to form the output sentence in natural language using the rounding process.

5 Fine Grained Style Transfer in Vision

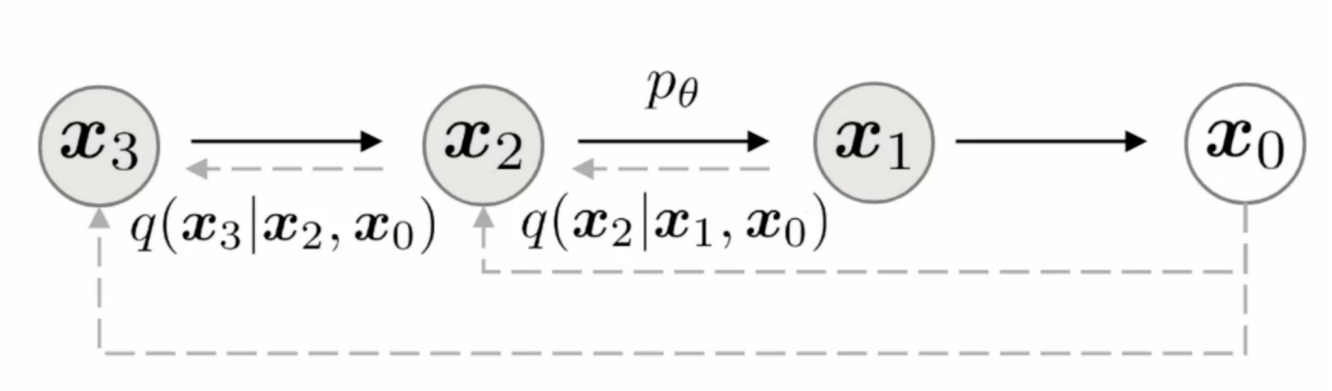
- In the domain of vision fine grained is achieved using the **STTR(STyle TRansformer)** architecture. We see the details about that and its inspiration from NLP. The **STTR** consists of four parts: (i) two backbones to extract and downsample features from the inputs (i.e., content and style images), (ii) two self-attention modules to summarize the style or content features, a (iii) cross-attention module to match style patterns into content patches adaptively, and (iv) a CNN-based decoder to upsample the output of cross-attention module to reconstruct the final result. We implement this by **Transformer** since its encoder consists of **self-attention** module while the decoder has **cross-attention** module to compute the correlations between **content** and **style tokens**.



- Now we will look into the about the main parts in the architecture:
  - **Tokenizer:** In this the image is divided into a set of visual tokens. Thus, we have to first convert the input image into a set of visual tokens. We assume that each of them represents a semantic concept in the image. We then feed these tokens to a Transformer.
  - **Encoder:** The encoder is used to encode style features, it consists of six encoder layers. Each encoder layer has a standard architecture and consists of a self-attention module and a feed-forward network (FFN).
  - **Decoder:** The decoder is used to reason relationships between content and style tokens while being able to model the global context among them. It consists of two multi-head attention layers and a feed-forward net- work (FFN). The first multi-head attention layer mainly learns the self-attention within content features while the second one learns the cross-attention between content and style features.
  - **CNN Decoder:** The CNN decoder is used to predict the final stylized image.
- From the architecture we can clearly see the **similarity** between our NLP setting and this architecture. **Now we plan to use this cross-attention architecture in our diffusion setting.**

6 DDIM for fast sampling

Now we see how DDIM sampling helps in sampling faster. The DDPMs are generalised via a class of non-Markovian diffusion processes that lead to the same training objective which correspond to generative processes that are deterministic, giving rise to implicit models that produce high quality samples much faster.



6.1 Defining the Family of Forward Stochastic Processes

- Now we consider the family of forward processes defined as :

$$q_\sigma(x_{t-1} \mid x_t, x_0) = \mathcal{N}\left(\sqrt{\alpha_{t-1}} x_0 + \sqrt{1 - \alpha_{t-1} - \sigma_t^2} \cdot \frac{x_t - \sqrt{\alpha_t} x_0}{\sqrt{1 - \alpha_t}}, \quad \sigma_t^2 \mathbf{I} \right) \tag{14}$$

- All the distributions are indexed by the real vector σ_t. The property of $q(x_t \mid x_0) = \mathcal{N}\left(\sqrt{\bar{\alpha}_t} x_0, (1 - \bar{\alpha}_t)\mathbf{I} \right)$ is still valid.

- For $\sigma_t = \sqrt{\frac{1-\alpha_{t-1}}{1-\alpha_t}} \sqrt{\frac{1-\alpha_t}{\alpha_{t-1}}}$ for all t, the forward process becomes Markovian, and the generative process becomes a DDPM.

- For $\sigma_t = 0$ for all t, the forward process becomes deterministic given x_{t1} and x_0. The resulting model becomes an implicit probabilistic model , where samples are generated from latent variables with a fixed procedure (from x_T to x_0).We name this the **denoising diffusion implicit model(DDIM)** because it is an implicit probabilistic model trained with the DDPM objective (despite the forward process no longer being a diffusion).

- Now we model the forward process as:

$$q_\sigma(x_t \mid x_{t-1}, x_0) = \frac{q_\sigma(x_{t-1} \mid x_t, x_0) q_\sigma(x_t \mid x_0)}{q_\sigma(x_{t-1} \mid x_0)}, \tag{15}$$

- Clearly the above forward process is no longer Markovian.

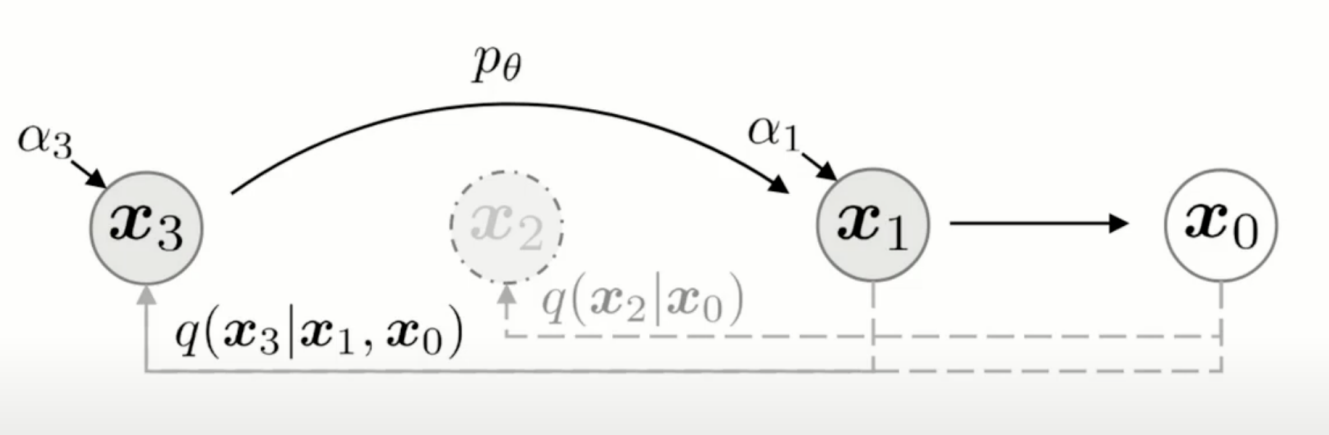
- After doing the required math for calculating the training objective we find that irrespective of the σ_t the training objective always simplifies to the training objective used in DDPMs.

- So this means that a model trained in the original DDPM process can be used by any of the process in the family for **inference**. Hence we use **DDIM** for sampling after training the **DDPM**.

- For sampling we do:

$$x_{t-1} = \sqrt{\alpha_{t-1}} \left(\frac{x_t - \sqrt{1 - \alpha_t}\, \epsilon_\theta^{(t)}(x_t)}{\sqrt{\alpha_t}} \right) + \sqrt{1 - \alpha_{t-1} - \sigma_t^2} \cdot \epsilon_\theta^{(t)}(x_t) + \sigma_t \epsilon_t, \tag{16}$$

6.2 Making Sampling faster



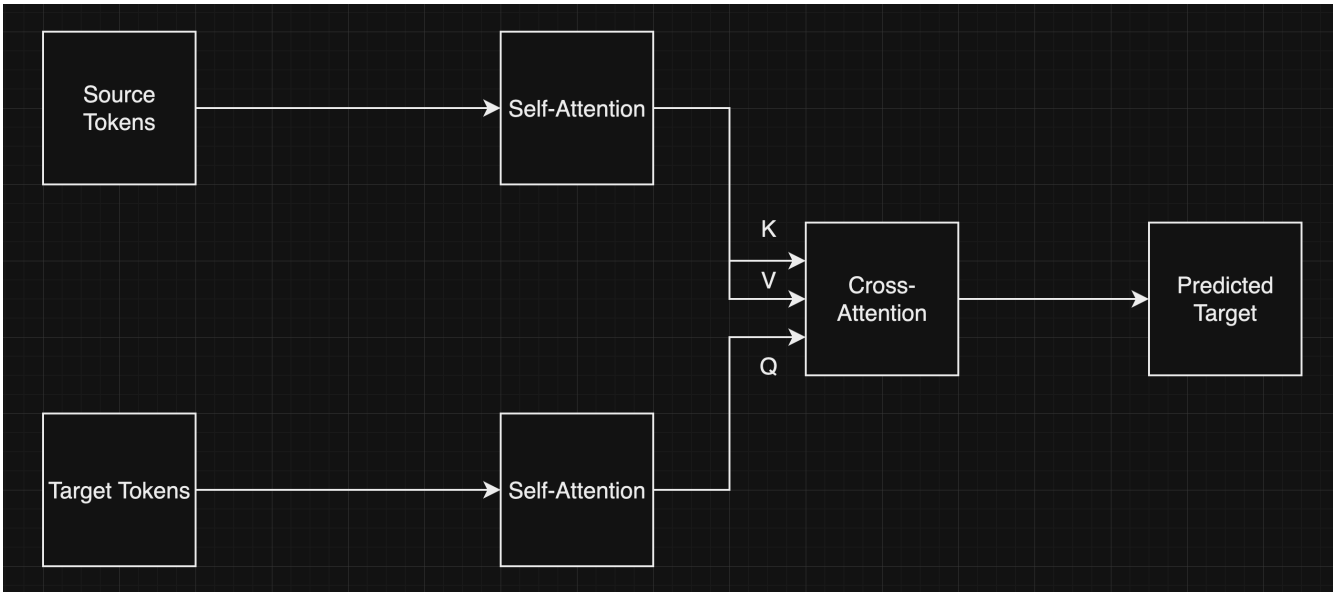
- Now we want to skip some steps while sampling for fast inference. That we show in the following steps that how this indeed matches with the **DDPM** training.
- Now defining only a subset of steps from $\{x_{\tau_1}, \ldots, x_{\tau_S}\}$.
- The backward process is defined as:

$$p_\theta(x_{0:T}) := p_\theta(x_T) \prod_{i=1}^{S} p_\theta^{(\tau_i)}(x_{\tau_i - 1} \mid x_{\tau_i}) \times \prod_{t \in \bar{\tau}} p_\theta^{(t)}(x_0 \mid x_t) \tag{17}$$

- After doing the required math the training objective for the above backward process is found to be equivalent with the original **DDPM**. So what we can summarise from this is that we can use the **DDIM sampling** with **skip steps** on a trained **DDPM**.

7 Our Proposal

7.1 Transformer Architecture



- We propose to use the above architecture based on the usage of cross attention in the domain of vision. Because of the closeness between the architecture setting of **STTR**(which is similar to NLP setting) and the setting of our current problem setting.
- We place the target tokens in the **decoder part** and the source tokens in the **encoder part**. Which is done based on the architecture of **STTR** by taking it as reference.
- We will be using eq(12) as our loss function.

7.2 For Sampling

We propose to use **DDIM Sampling** during inference.

8 Evaluation Criteria

- The **BLEU** score is calculated as follows:

$$\text{BLEU} = \text{BP} \cdot \prod_{n=1}^{N} p_n^{w_n} \tag{18}$$

- p_n is the $N - gram$ and is calculated as follows:

$$\text{Precision} = \frac{\#\text{overlapping words}}{\#\text{predicted words}} \tag{19}$$

- BP is called **Brevity Penalty** which is defined as follows:

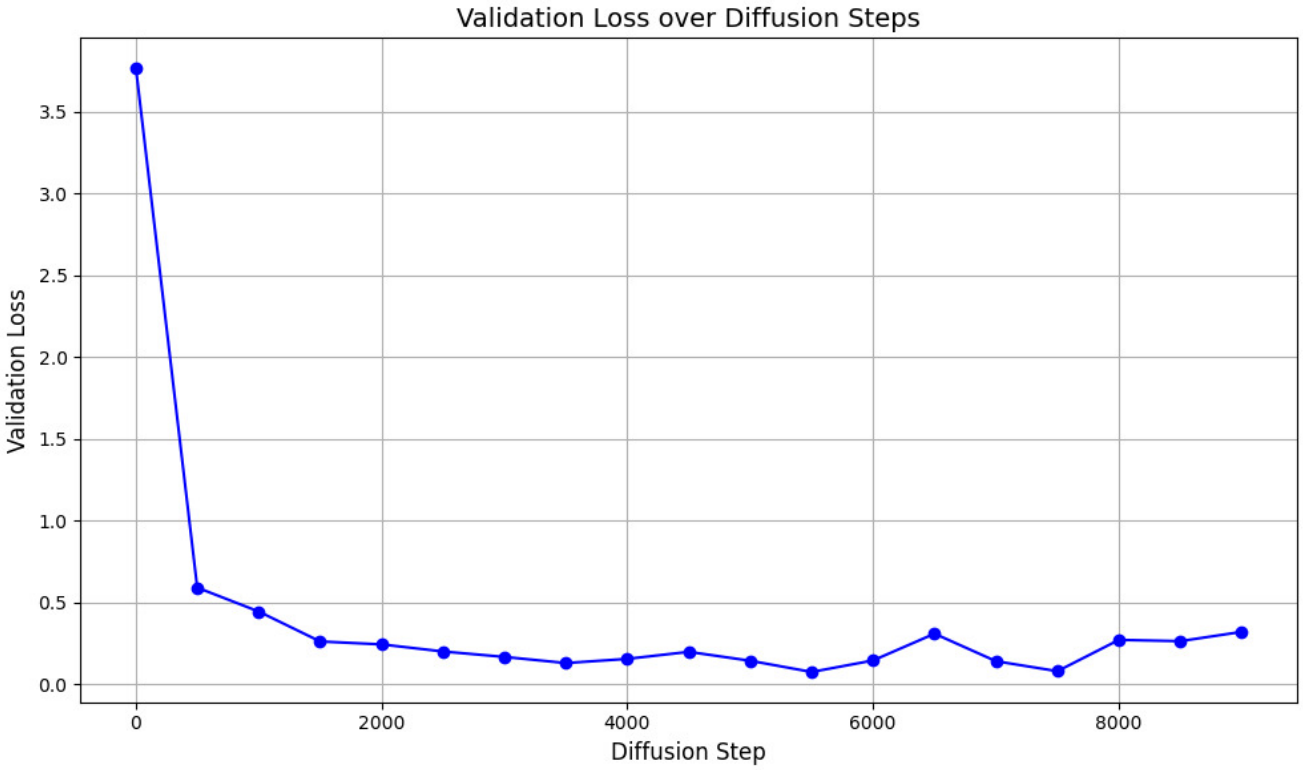
$$\text{Brevity Penalty(BP)} = \begin{cases} 1, & \text{if } c > r \\ e^{(1-r/c)}, & \text{if } c \leq r \end{cases}$$

  where:

  - c is *predicted length* = number of words in the predicted sentence
  - r is *target length* = number of words in the target sentence

9 Results

- We have only trained on the **PPR(Pre-Position Removal)** task. This is a **Medium level** task.
- Hyperparams choice:
  - **learning rate** : 10^{-4}
  - **batch size** : 64
  - **noise schedular** : linear
  - **sequence length** : 128
  - **learning steps** : 9000



- The above is the graph for training the proposed model. We have conducted the experiments based on the models checkpoint at learning step **5500**.

Our Outputs:

Source	Target	Prediction
these securities are attractive to japanese investors for three reasons	these securities are attractive	entourage [PAD] recover madden [PAD] numb [PAD] [PAD] sal [PAD] [PAD] bearingsgood [PAD] [PAD] [PAD] [PAD]lina voltage hurt [PAD]uesumatic [PAD] [PAD] [PAD] [PAD] [PAD] 9 [PAD] [PAD] [PAD] hahn grossed [PAD] comfortably signage dazzling measurement [PAD] [unused688] [PAD] nilsson bus rosewood is row [PAD] [PAD] futures bothered [PAD] [PAD] recaptureduri [PAD] [PAD] [PAD] [PAD] pay shuffling leo [PAD] [PAD] [unused196] [PAD] [PAD] dimective abolition objectionscarriagezily peacefully [PAD] [PAD] [PAD] [PAD] [PAD] jo [PAD] [PAD] [PAD] [PAD] fated [PAD] stoke var [PAD] [PAD] [PAD] [PAD] comprises [PAD] ut [PAD]yler unanimously [PAD] [unused536] [PAD] subtle reflectedders [PAD] [PAD] commencement solomon cappedible milli
the unk auto maker last year sold NUM cars in the u.s	the unk auto maker last year sold num cars	cold gettysburg speedy normally [PAD] [PAD] harta [PAD] [PAD] enclosure approvedmeral auckland strong [PAD] calculating infringement shine [PAD] [PAD] [PAD] [PAD] [PAD] [PAD] 650 [PAD] [PAD] [PAD] promise [PAD]town 274 tragic [PAD] digitaltrip [PAD] [PAD]bl [PAD] [PAD] muttering 280 [unused499]chase [PAD] [PAD] [PAD] santana belief [unused610] [PAD] enoch clubhouse [unused363] [PAD] [PAD] [PAD] diversityaver [PAD] [PAD] stupid bowenrs [PAD] [PAD] [PAD] [PAD] [PAD] [PAD] steppeieving dean melbournetsk revoked overs [PAD] [PAD] [PAD] correspondent [PAD] [PAD] [PAD]tze nuts import [PAD] lenny [PAD] culturezily sanskrit brodie [unused329]bry [PAD] [PAD] [PAD] [PAD] source g
the dollar weakened against most other major unk	the dollar weakened	[PAD] [PAD] [PAD] helmets t elves [PAD] deadline dukes [PAD] [PAD]² [PAD] [PAD] [PAD] [PAD] gabe veronalingerhl [PAD]lwyn [unused550] [PAD] [PAD] [PAD] chevalier [PAD] [PAD]² 48th [PAD] [PAD]jian [PAD] sizes [PAD] helmets [PAD] [PAD] [PAD] [PAD] nano [PAD] [PAD]zano [PAD] [PAD] [PAD]wehr hurt [PAD] [PAD] harmonica [PAD] [PAD] citizen [PAD] [PAD] [PAD] carrying [PAD] [PAD] heaviest [PAD] colonies [PAD] [PAD] [PAD] [PAD] laptop [PAD] [PAD] clasp [PAD] [PAD] [PAD]nging [PAD] [PAD] 9 spoke [PAD] [PAD]inates affordable [PAD] [PAD] [PAD] pounder [PAD] wong [PAD] [PAD] snorted [PAD] trouble [PAD] regannberg recover [PAD] [PAD] consuming terms formationsmarine dow [PAD] michel

References

[1] Understanding Diffusion Models: A Unified Perspective

[2] DiffuSeq: Sequence to Sequence Text Generation with Diffusion Models

[3] Fine-grained Text Style Transfer with Diffusion-Based Language Models

[4] Fine-Grained Image Style Transfer with Visual Transformers

[5] Denoising Diffusion Implicit Models

[6] StylePTB: A Compositional Benchmark for Fine-grained Controllable Text Style Transfer