Paraphrase Diversification using Counterfactual Debiasing

Sunghyun Park[1,2], **Seung-won Hwang**[1*], **Fuxiang Chen**[2,4],
Jaegul Choo[3], **Jung-Woo Ha**[2], **Sunghun Kim**[2,4], **Jinyeong Yim**[2]
[1]Yonsei University [2]Clova AI Research, NAVER [3]Korea University [4]Hong Kong University of Science and Technology

Abstract

The problem of generating a set of diverse paraphrase sentences while (1) not compromising the original meaning of the original sentence, and (2) imposing diversity in various semantic aspects, such as a lexical or syntactic structure, is examined. Existing work on paraphrase generation has focused more on the former, and the latter was trained as a fixed style transfer, such as transferring from positive to negative sentiments, even at the cost of losing semantics. In this work, we consider style transfer as a means of imposing diversity, with a paraphrasing correctness constraint that the target sentence must remain a paraphrase of the original sentence. However, our goal is to maximize the diversity for a set of k generated paraphrases, denoted as the diversified paraphrase (DP) problem. Our key contribution is deciding the style guidance at generation towards the direction of increasing the diversity of output with respect to those generated previously. As pre-materializing training data for all style decisions is impractical, we train with biased data, but with debiasing guidance. Compared to state-of-the-art methods, our proposed model can generate more diverse and yet semantically consistent paraphrase sentences. That is, our model, trained with the MSCOCO dataset, achieves the highest embedding scores, .94/.95/.86, similar to state-of-the-art results, but with a lower mBLEU score (more diverse) by 8.73%.

1 Introduction

Paraphrasing is the task of rephrasing a given sentence into another with the same semantic meaning. Related tasks include paraphrase identification, classifying whether the given pair of sentences is a paraphrase of each other. Another task is to generate a paraphrase sentence of a given input sentence. Both tasks are useful in numerous NLP tasks, including question answering (QA) and information retrieval (IR). For example, a new question can be a paraphrase of an existing QA pair, which can be retrieved as an answer to a new question. Similarly, in IR, a part of a document that matches the query as its paraphrase can be a good retrieval candidate. However, in both scenarios, we often fail to find paraphrase pairs, due to lexical and structural differences (Zhou et al. 2015).

*corresponding email:seungwonh@yonsei.ac.kr

Existing paraphrasing work mainly focuses on preserving the original sentence's semantic meaning but less on generating diversified paraphrases (see Section 2.3). This study addresses the generation problem, with a focus on diversification. With the advent of sequence-to-sequence (Seq2Seq) models, after encoding the given sentence, paraphrase can be decoded in various forms from a given encoded sentence, by adding random noise to deep generative models, such as a variational autoencoder (VAE) (Kingma and Welling 2014). However, these models reportedly have common weaknesses. The generated sentences, to preserve the semantic meaning, are often safe yet tedious repetitions of the original sentence, as frequently observed in generative models (Xu et al. 2018). Our objective is to generate a set of paraphrases $p_1, \ldots, p_k$ to maximize diversity.

Another closely related task is a style transfer that rephrases a given sentence to impose a particular style property. One can view the style transfer satisfying paraphrase identification as a special paraphrase case. This property can be implicitly trained using a paired or an unpaired corpus (Fu et al. 2017) or in an explicitly guided manner (Iyyer et al. 2018), such as enforcing the syntactic structure of a paraphrase.

While style transfer focuses on imposing a single fixed form of diversity, we extend that to paraphrase diversification for a generated set of paraphrases $p_1, \ldots, p_k$, given the original sentence o and its paraphrase p, as well as the generated paraphrase history. When generating the first paraphrase (without any history), history is set as null. Specifically, we generate an uncontrolled random noise z, then transformed it into z' to provide diversity guidance during generation.

Due to the challenging fact in pre-materializing the existing paraphrase training data to include all possible diversity guidance during generation, our factual training instances are often biased (e.g., having syntax that is too similar). Training using counterfactual debiasing instances (having dissimilar syntax) thus pave way to allow for more diversity guidance. To allow that, in this work, we provide a debiasing guiding vector for generating counterfactual (debiased) training instances. We summarize our key contributions as follows.

- Random noise z is uncontrollable such that a particular style property cannot be imposed; so, it is transformed

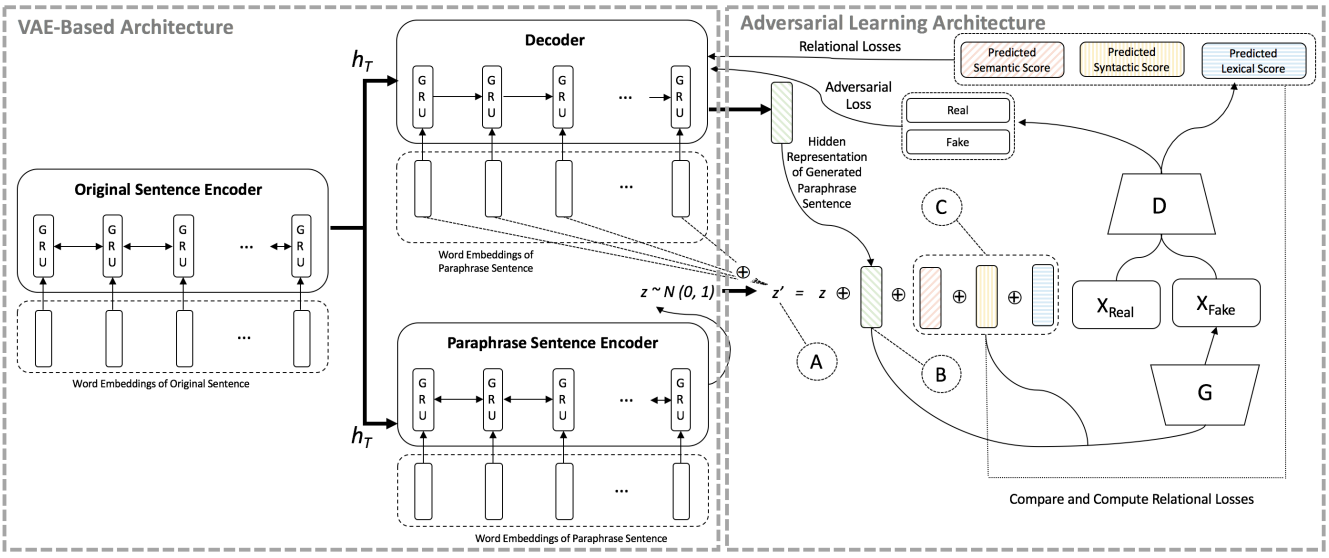

Figure 1: Overview of the proposed approach, combining a VAE-based and adversarial learning architectures that mutually communicate for the purpose of extracting different styles and discriminating the set of generated paraphrases that are dissimilar from the original sentence, respectively. In addition, we consider the previously generated paraphrase as another diversity condition in generating paraphrases. **A** shows that our controlled latent vector z' is concatenated with each of the word embeddings in the paraphrase during decoding. In **B**, the hidden representation of the previously generated paraphrase (from the decoder) is concatenated with z from the current paraphrase-sentence encoder. In the first step, the hidden representation of the previously generated paraphrase is initialized as all zeros. **C** represents the semantic, syntactic, and lexical similarity vectors (left to right) between the original sentence and its paraphrase.

into z' to reflect our debiasing guidance.

- We use an aggregated representation of previous paraphrase generations as a vector, so that the output is guided to be different from that previously generated.
- We overcome the lack of quantity and diversity in training data, by leveraging negative pairs and adversarial examples for training.

2 Related Work

2.1 Paraphrase Generation and Identification

Due to page limitation, we refer the reader to (Androutsopoulos and Malakasiotis 2010) for a comprehensive survey of paraphrase generation and identification methods; here, we focus on the "diversity" aspects of this work.

Recent Seq2Seq paraphrasing models pursue diversity primarily in two directions. First, decoding approaches diversify outputs using techniques such as top-k beam search (Gupta et al. 2018). Second, random noise is introduced in a generator as an additional input (Dai and Lin 2017; Jain, Zhang, and Schwing 2017). The weakness of the former is that the quality of the output degrades as the number of the outputs grows (or when the lower-ranked results from the beam search are presented). The second group addresses such weaknesses by generating the top-1 result for k different random noises. However, the degree of diversity is often marginal, as we have no control over z.

Another line of related work involves the hypothesis that paraphrase generation is a sequence of edit operations applied to the original sentence (Huang et al. 2018; Guu et al. 2018; Li et al. 2018; Quirk, Brockett, and Dolan 2004). Compared to Seq2Seq models in generating sentences from scratch, this approach deals with a smaller and discretely guided search space in generating paraphrases but in return, yields higher efficiency and quality throughput. However, such a hypothesis limits diversity by assuming the generation of a paraphrase is conditioned on the original sentence by some set of edit operation. Our work has the benefit of a guided search space, using a guide vector, but without making such a hypothesis.

Paraphrase identification, classifying whether the given two sentences are paraphrases of each other, has a richer history of study (Androutsopoulos and Malakasiotis 2010). We focus on discussing how such work contributes to guiding paraphrase generation. We can train *evaluator*, which is a paraphrase identification module, to judge whether the given and the generated sentences are paraphrases of each other. Networks of such evaluators can guide generator networks, either through reinforcement learning (Li et al. 2017), or through adversarial training (Su et al. 2018). Our work is an example of adversarial training between a generator and a discriminator.

2.2 Style Transfer

Similar to machine translation of text written in one language to another, image translation has recently been a popular topic, converting the style of a given image into another, e.g., (Johnson, Alahi, and Fei-Fei 2016). Leveraging an adversarial training framework, style transfer techniques have been significantly improved (Zhu et al. 2017). In general, these early studies require a training set of image pairs composed of an original input and its output image after the style transfer. Similarly, in the domain of natural language generation, the early work on style transfer in text builds on parallel resources. For example, (Jhamtani et al. 2017) converts modern text into Shakespearean English using paired data. Separately, in the work of (Prabhumoye et al. 2018; Xu et al. 2012), paraphrases are generated by transferring different types of styles, such as gender, political slant, and ancient writing, into the original sentences. However, all these studies address the problem of fixed styles. For automatic generation of training resources, back-translations between a source language and a target language (Lample et al. 2018; Artetxe et al. 2018) has been used for paraphrases (Wieting and Gimpel 2017). However, back-translations as a training resource is not suitable for diversity, as their syntactic variation is reportedly limited (Iyyer et al. 2018).

Instead of implicitly defining a style through paired and unpaired data as above, an explicit guidance, such as specifying a syntactic structure of the paraphrase (Iyyer et al. 2018), has also been studied. We leverage both approaches by explicitly identifying which aspect to diversify (based on the given input), while implicitly learning style representation, through adversarial training, learning to separate content and style representations, as similarly used in (Fu et al. 2017).

In summary, our work is distinguished as follows:

- **Semantic preserving:** Unlike some style transfer, such as turning positive statements into negative, semantic preservation is essential for paraphrase correctness.
- **Debiasing guidance:** Unlike the goal of maintaining consistency to one style, for which training data is naturally biased to one style, style changes based on previous generation in our problem, for which proper training instances may not exist. We thus present a biased training data, with unbiasing guidance, to train on the debiased counterfactual data. Our framework can also subsume a classical problem of generating k paraphrases with a fixed style, simply by not providing any such guidance.

2.3 Diversification

Lack of diversity in generative models has been a major issue in natural language generation, such as conversation (Xu et al. 2018) or paraphrasing (Iyyer et al. 2018), as we have already discussed above. However, their diversity requirement is applied within a pair of sentences. To pursue set diversity, we can consider a search engine result diversification task – For a given query of keywords, matching multiple semantic meanings, pursuing only a traditional goal of relevance would provide the top k results covering only a dominant one. A diverse set would generate a set that includes a minority sense as well, without compromising much relevance (Jiang et al. 2017).

3 Proposed Approach

Our goal is to generate a set of diverse paraphrases for each input pair, composed of the original sentence and its paraphrase. First, we provide an overview of the VAE-based (SOTA) approach, which utilizes a latent vector z (Section 3.1). Afterwards, we gradually introduce noise in two steps into a controlled latent vector z' (Sections 3.2 and 3.3), which will be fed into a variational autoencoder (with the guidance of a discriminator in Section 3.4), so that the autoencoder can generate diversified paraphrases, conditioned upon z'.

3.1 VAE-based Paraphrase Generation

This section illustrates a VAE-based state-of-the-art paraphrase generation model (Gupta et al. 2018), shown on the left-hand side of Figure 1, as one of our baselines. This model only considers z but not z' (Figure 1).

Given a pair of original sentence $o = (w_{o_1}, w_{o_2}, ..., w_{o_n})$ and its paraphrase $p = (w_{p_1}, w_{p_2}, ..., w_{p_n})$, let us denote the word embeddings of the two as $e_o = (e_{o_1}, e_{o_2}, ..., e_{o_n})$, and $e_p = (e_{p_1}, e_{p_2}, ..., e_{p_n})$, respectively. The word embeddings of the original sentence and its paraphrase are encoded using bidirectional-GRU encoders (Chung et al. 2014) (shown as the inputs to the *Original Sentence Encoder* and *Paraphrase Sentence Encoder* in Figure 1, respectively). In addition, the word embeddings of the paraphrase are also passed to a GRU decoder (shown as the inputs to *Decoder* in Figure 1).

The hidden representation of the original sentence (through the *Original Sentence Encoder*), h_T, is fed as an additional input to both *Paraphrase Sentence Encoder* and *Decoder*. A latent vector z is then randomly sampled from the mean and variance vector representations of the *Paraphrase Sentence Encoder*.

Compared to classical LSTM decoder models, which use a beam search to generate multiple candidates, VAE-based models have the strength of generating multiple different hidden representations z's to generate diverse candidates with comparable quality. However, with uncontrolled z for the pairwise paraphrase generation objective, we need to overcome three new challenges, as follows:

C1: Unlike uncontrollable z, which cannot guide a style property, we need to generate z' that reflects our guidance.

C2: We need to represent previous paraphrase generations to impose diversity as compared with those previously generated.

C3: We need to overcome the lack of quantity and diversity in training data, by leveraging negative pairs and adversarial examples for training.

3.2 C1: Generating Controlled z' from Guidance

We first discuss how to perturb the original latent vector z into a controlled vector z', which has a different style.

To our knowledge, the closest study is (Iyyer et al. 2018), which specified a syntactic structure to ensure the lexical

diversity on the paraphrase sentences. However, we empirically observe that such generation, being constrained for the explicit syntax requirement, often violates the semantic equivalence required for the generated paraphrase.

Our goal is to explicitly guide the perturbation of z into z', to encourage the diversity, specifically, with respect to the following three dimensions: $Semantic(Sem)$, $Syntactic(Syn)$, and $Lexical(Lex)$.

We first train a classifier that takes a pair of sentences as input and returns a three-dimensional binary vector, denoted as a guide vector, [Sem, Syn, Lex]. Sem is set to one, if the pairs are labeled as a paraphrase, and 0 for a negative example. Inspired by (Iyyer et al. 2018), Syn compares the parsed results of the two sentences and quantifies their similarity into a binary score. [1] Lex indicates whether the unigram Jaccard similarity (Goodall 1966) of a given pair is higher than 0.5.

We then train a controlled representation z', using a concatenation of an embedded labeled vector $[e_{Sem}; e_{Syn}; e_{Lex}]$ and z,

$$z' = \sigma(w^T[z; e_{Sem}; e_{Syn}; e_{Lex}]) \quad (1)$$

where w is the parameter learned to separate the opposite labels as much as possible.

To illustrate, when the given paraphrase pair has a similar lexical structure, a guide vector will indicate such similarity to encourage lexical diversity when generating z'.

3.3 C2: Set-Diversification via Previous Generated Paraphrases

We now extend the notion of pairwise diversity from the original sentence into diversity within a set of generated paraphrases: Existing models generate paraphrases without considering the previously generated sentences (Xu et al. 2017; Li et al. 2015). As a result, a set of paraphrases generated from a single original sentence tends to be highly similar with one another.

In contrast, we utilize the previously generated paraphrases to generate an additional paraphrase that is different. In detail, we first concatenate the hidden representation, h_g, from its previous paraphrase to z' and introduce a novel loss term, called a *word coverage loss*, by computing the word level differences between the generated paraphrase and its previous version.

In *word coverage loss*, the word-level differences between the generated paraphrase and its previous paraphrases is computed by first generating two different vectors, corresponding to the generated and previous paraphrases Gp and Pp respectively. The dimension of the vectors equals the size of the vocabulary. If a word is present in Gp or Pp, the vector cell corresponding to the vocabulary will be marked as 1, otherwise, 0. We then subtract Pp from Gp in returning a single vector. The *word coverage loss* is then computed by summing the cells of the vector that contains the value 1.

Finally, the latent vector z' is represented as

$$z' = \sigma(w^T[z; e_{Sem}; e_{Syn}; e_{Lex}; h_g]). \quad (2)$$

[1] We follow the convention of (Iyyer et al. 2018) to use the top two-level parses.

3.4 C3: Diversity beyond Training Data

Our last challenge is to confirm that a guided paraphrase can be generated even when such transfer is not frequently observed in the training dataset.

This is important because lexical and syntactic diversity is limited in many public datasets. For example, in the Quora dataset, only 10% of the original and paraphrase questions are lexically diverse. Desirably, our guided vector, even when given the pair without lexical diversity, should still enforce the generation of a set of diverse paraphrases. That is, if Syn and Lex suggest that the given pair lacks diversity in either aspect, we can set the guidance vector as its negation, $\overline{Syn}$, $\overline{Lex}$, to enforce the diversity (or negated guidance) requirement. For training a negated effect, we use negative sampling, to be elaborated further.

This section discusses how we guide the generation of a set of diverse paraphrases through discrimination of the generated paraphrases from the original sentences. Specifically, we train an adversarial network, consisting of a generator (*G*) and a discriminator (*D*) (shown on the right side of Figure 1). The generator is used to produce paraphrases that are similar to the training data (human-written paraphrases), while the discriminator learns the similarity [Sem, Syn, Lex] between the generated paraphrases and the original sentence to further guide the generator in recognizing and producing human-like written paraphrases.

We describe the two loss functions used in our adversarial network:

(1) *GAN Adversarial Loss* (Goodfellow et al. 2014)

$$\begin{aligned} L_{adv} =& \mathbb{E}_p \left[D_{adv} \left(p \right) \right] \\ &+ \mathbb{E}_{o,p,c} \left[\log \left(1 - D_{adv} \left(G \left(o, p, c \right) \right) \right) \right] \end{aligned} \quad (3)$$

(2) *Relational Loss* – In addition to distinguishing between human-written and machine-generated paraphrases, the adversarial network also learns to distinguish the [Sem, Syn, Lex] features between the human-written and machine-generated paraphrases. We call the loss of this function the *Relational Loss*. Mathematically, the relational loss function is shown as the following:

$$\begin{aligned} L_{rel} =& \mathbb{E}_{o,p,c} \left[- \log D_{rel} \left(c | o, p \right) \right] \\ &+ \mathbb{E}_{o,p,c} \left[- \log D_{rel} \left(c | o, G \left(o, p, c \right) \right) \right] \\ &+ \mathbb{E}_{o,p,\bar{c}} \left[- \log D_{rel} \left(\bar{c} | o, G \left(o, p, \bar{c} \right) \right) \right] \end{aligned} \quad (4)$$

The original sentence and its paraphrase are denoted as o and p, respectively, while c denotes the similarity condition (i.e., Sem, Syn, Lex). The use of the negative sampling per similarity condition is given by $\bar{c}$ (i.e., $\overline{Syn}$, $\overline{Lex}$). As the generated paraphrases should be semantically similar to the original sentence, we do not perform a negative sampling on Sem. The second and the third parts of equation 4 represent the use of positive and negative sampling in calculating our proposed relational loss function respectively.

We note that the use of negative sampling for paraphrase generation has not been attempted in previous studies, and our work is the first to introduce negative sampling for paraphrase generation.

4 Experimental Setup

We describe the details of the model's decoder (Section 4.1), the datasets used for training (Sections 4.2 and 4.3), and the evaluation of the generated paraphrases, both quantitatively, through different metrics (Section 4.4), and qualitatively, through a user study (Section 4.5). We implemented our models and experimented using NSML (Kim et al. 2018).

4.1 Details of the Decoder

In our decoder networks, we use a beam search, in which the beam size is set to 10, following the same setting as the previous SOTA model (Gupta et al. 2018), and we did not perform any additional reranking.

4.2 Ensemble of Skewed Transfers

To illustrate the robustness and effectiveness of the models on different types of data, we used multiple paraphrase datasets for training/development/testing (Quora 2018; Bowman et al. 2015; Dolan, Brockett, and Quirk 2005; Lin et al. 2014; Coster and Kauchak 2011). These datasets are also widely used in previous paraphrase generation work (Prakash et al. 2016; Gupta et al. 2018; Brad and Rebedea 2017). We note that our work is the first to use multiple different datasets for evaluation. We describe the datasets used in the following:

Quora: Released by Quora in 2017, this dataset contains question pairs (asked in Quora) that are paraphrases of one another. It consists of 400K pairs of questions, in which 140K are annotated as paraphrased questions, while the others are not (Quora 2018). We used the sentence pairs that are labeled as duplicates/paraphrases as the paraphrase dataset.

Microsoft: Released by Microsoft in 2005, this dataset consists of 5,800 pairs of sentences extracted from online news sources that are annotated as paraphrases.

SNLI: Released by the Stanford NLP Group (Stanford 2018) in 2015, this dataset contains 570K of human-written English sentence pairs that are human-labeled as entailment, contradiction, and neutral, in which each group of sentence pairs totals approximately 190K. We used the sentence pairs that are labeled as entailment as the paraphrase dataset.

MSCOCO: Released by COCO Consortium (COCO 2018) in 2017, this dataset consists of 123K images that are human-annotated with five annotations per image (Lin et al. 2014). In terms of generating a sentence paraphrase pair, we follow the strategy of (Gupta et al. 2018), i.e., randomly removing one of the five image annotation, and combining the other four image annotations into two different sentence paraphrase pairs.

Wikipedia: Released by Coster and Kauchak (Coster and Kauchak 2011) in 2011, this dataset consists of 137K of sentences from Wikipedia and its simplified form.

Combined We further combined all the above five different sources of paraphrase datasets into a separate larger group of paraphrase datasets for evaluation.

For each of the above six datasets, we randomly split the dataset into training/development/testing, following the distribution of previous studies (Gupta et al. 2018; Patro et al. 2018). None of the data from the training, development, or testing sets are observed to be the same. The trained models are tuned based on the development dataset. For testing, we combined all the test data of the dataset into a common test set, and use it to test every trained model.

4.3 Non-Paraphrase Datasets

As mentioned in Section 3.4, we combined non-paraphrased datasets with the paraphrased versions and use them to train separate models for evaluation. Similar to Section 4.2, for Quora and SNLI, we used the non-duplicate question pairs (from Quora) and the question pairs marked as neutral or contradiction (from SNLI). This accounts for approximately 60% of the total paraphrase sentence pairs. We randomly sampled this *negative set* and combined with the *positive set* (paraphrased sentences) for training. The sample size of the *negative set* is the same as the *positive set* that is used for training the paraphrase datasets (*positive set*).

For the other datasets, we generate the non-paraphrase (*negative*) datasets by randomly selecting two pairs of original-paraphrase sentences and exchanging their paraphrase sentences. We repeat this process until we have the same size of (*positive set*) used to train each model.

4.4 Metrics and Baselines

We categorize metrics for measuring similarity (lexical, semantic, and structural) and diversity. We note that there is a tension between similarity/diversity measures and that no single solution is the winner in all aspects (i.e., they are closely dependent on one another).

Lexical Similarity

BLEU (Papineni et al. 2002): This metric quantifies the lexical similarity of the generated paraphrases to the human references, by counting the common n-grams. A low BLEU score indicates that the generated paraphrase does not closely resemble the human-written ground-truth paraphrase. However, the BLEU score may not perfectly represent semantic similarity, as "autumn leaves" and "fall foliage" will get a low BLEU score, given the former as a reference. We still use this metric due to its popularity.

Semantic Similarity

Embedding Similarity: This metric measures the word embedding differences between the generated and the ground-truth paraphrase sentences. Following the work of (Xu et al. 2017), we also use average, extreme, and greedy (A/E/G) embedding differences.

METEOR (Denkowski and Lavie 2014): This metric measures the alignment between the generated and the ground-truth paraphrase sentences by exact, stem, synonym, and paraphrase matches between words and phrases.

A high embedding similarity score in (A/E/G) and a high METEOR score indicate that the generated paraphrases have similar meaning when compared with the original sentence.

Structural Similarity

Parse Tree Similarity: We follow (Iyyer et al. 2018) in cal-

culating the top two levels of parse tree similarity among the generated paraphrases.

Diversity (among generated paraphrases)
Dist-n (Li et al. 2015): This metric measures the number of distinct n-grams within the set of generated paraphrases, denoted as *Dist-n*. Following previous studies (Xu et al. 2018; 2017), we use Dist-1, 2, and 3.
mBLEU (Fan et al. 2018): This metric computes the dissimilarity of *BLEU* scores within the set of generated paraphrases. For example, each generated paraphrase will be compared with other generated paraphrases in terms of *BLEU* scores to compute an average *BLEU* score.

A high Dist-1/2/3 score and a low mBLEU score indicate that the generated paraphrases are diverse among themselves.

Baselines
We evaluate six different models in which each of them is trained with six different datasets as described in Sections 4.2 and 4.3. These models are Seq2Seq (with attention and beam search), state-of-the-art model (Gupta et al. 2018), and two of our models (one trained with positive paraphrases only, and the other with both positive and negative paraphrases). We also incorporated a history module that takes into consideration the previous generated paraphrases, which are concatenated with the original input sentence in the encoder, in Seq2Seq, and (Gupta et al. 2018) for additional comparison. All models are configured consistently to produce the top eight generated paraphrases [2]. Eight generated paraphrases are chosen for all models for fair comparison. We compare eight generated paraphrases as our model consists of lexical, structural, and history properties, and we enumerate and consider all possibilities among the lexical, structural and history choices, while maintaining the semantics.

4.5 User Study

We sent an online advertisement to all graduate students from a university, and 21 of them participated in our study. The users were given a set of 20 randomly selected original sentences, together with their eight generated paraphrases, from the best of Seq2Seq and (Gupta et al. '18), including the ones with the history module, Ours_{pos}, and Ours_{all} models (having the highest semantics similarity). The users were required to choose the set of paraphrases that illustrates the highest semantic similarity and diversity when compared with the original sentence.

5 Results

5.1 Quantitative Analysis

Table 1 shows the quantitative evaluation of the different models. The first and the second columns show the different models used for evaluation and their lexical similarity, respectively. Semantic similarity scores are shown in the third and the fourth columns while structural similarity is shown in the fifth column. Diversity scores are shown in the sixth and the seventh columns. The lexical and semantic similarity metrics are compared against the ground-truth paraphrase sentences, while the structural similarity and diversity metrics are compared among the generated paraphrases.

In terms of diversity, our models perform better when trained with Wikipedia and Quora datasets, using both positive paraphrase sentence pairs (Ours_{pos}) and combined positive and negative paraphrase sentence pairs (Ours_{all}) respectively. We note that the diversity scores (Dist-n and mBLEU) of our models outperform both SOTA and Seq2Seq models. Specifically, $\text{Ours}_{\text{all (Quora)}}$, our best model in terms of diversity, surpassed the best SOTA (Gupta et al. '18 (Wikipedia)) and Seq2Seq (Seq2Seq (SNLI)) models with mBLEU scores of 20.88 and 20.36, respectively. We also observed that adding the history module in Seq2Seq and (Gupta et al. 2018) does not help in enhancing their diversity.

For structural similarity, although Seq2Seq (Microsoft) has the lowest parse tree similarity among all the generated paraphrases (i.e., 0.39), we note that, on average, Seq2Seq has the highest mean parse tree similarity score, followed by (Gupta et al. '18), Ours_{pos}, and Ours_{all}. We also observe that Seq2Seq (Microsoft), despite having the lowest structural similarity score, has the lowest semantic similarity scores.

There is a common pattern that all models trained either with the Combined or MSCOCO dataset perform better in terms of semantic (*Embeddings* and *METEOR*) and lexical (*BLEU*) similarity. We believe this is due to its larger dataset size. When comparing both semantic similarity and diversity, i.e., models trained with the Combined or MSCOCO dataset, our model, Ours_{all}, has the best Dist-n and mBLEU scores, outperforming Seq2Seq, (Gupta et al. '18) and Ours_{pos} by 0.7, 8.73 and 8.33, respectively. We further note that for the Seq2Seq model, its semantic similarity scores are much lower than Ours_{all}, and for (Gupta et al. '18) and Ours_{pos}, they have similar scores.

Overall, our best model, $\text{Ours}_{\text{all (MSCOCO)}}$, when compared to all other models, including SOTA, has the highest semantic similarity with the ground truth, and yet among its set of eight generated paraphrases, they show the highest diversity. In addition, on average, Ours_{all} has the smallest structural similarity score among its generated paraphrases.

5.2 Qualitative Analysis

In Table 3, we report the user study results. The first column shows the different types of models that generate the paraphrases, and the second and third columns show the percentage of users who select the set of generated paraphrases that best describe the highest semantic similarity and diversity when compared to the original sentence. A majority of the users selected Ours_{all} as the best model, followed by Ours_{pos}, Gupta et al. '18_{history}, $\text{Seq2Seq}_{\text{history}}$, Seq2Seq and Gupta et al. '18.

Table 2 shows the generated paraphrases of each model that output the highest diversity (mBLEU) score. The first two rows show the original sentence and the ground-truth

[2]This covers eight combinations of three binary conditions for with and without 1) generation history, 2) lexical debiasing, and 3) syntax debiasing

Table 1: Quantitative evaluation of generated paraphrases. Our best models, Ours_{pos} and Ours_{all}, outperformed other baseline models, including SOTA, in terms of diversity (Dist-1/2/3 and mBLEU). Specifically, the training of negative paraphrase sentence pairs, which we introduced in our model (Ours_{all} (Quora)) outperformed SOTA (Gupta et al. '18 (Wikipedia)) and Seq2Seq (Seq2Seq (SNLI)) by margins of 20.88 and 20.36, respectively, in the mBLEU score. Comparisons in both semantics and diversity also reveal consistent outperformance vs. SOTA and Seq2Seq in terms of generating paraphrases with high semantic and diversity scores.

Model	*Lexical Sim*	*Semantic Sim*		*Structural Sim*	*Diversity*	
	BLEU	**Embeddings (A/E/G)**	**METEOR**	**Parse Tree Sim**	**Dist-1/2/3**	**mBLEU**
Seq2Seq (Quora)	1.58	.78/.84/.41	3.99	0.86	.0009/.009/.02	60.09
Seq2Seq (Microsoft)	0.03	.75/.78/.47	2.09	**0.39**	.00004/.0002/.0007	65.95
Seq2Seq (SNLI)	3.36	.85/.89/.47	8.92	0.78	.0007/.006/.01	**54.67**
Seq2Seq (MSCOCO)	**5.96**	**.88/.91/.50**	11.76	0.69	.0004/.002/.007	55.92
Seq2Seq (Wikipedia)	3.73	.82/.87/.43	7.47	0.72	**.001/.03/.09**	61.19
Seq2Seq (Combined)	5.66	.88/.90/.52	**11.78**	0.71	.0003/.002/.006	59.07
Gupta et al. '18 (Quora)	6.49	.85/.88/.56	13.36	**0.47**	.001/.07/.24	62.28
Gupta et al. '18 (Microsoft)	2.28	.84/.86/.48	8.85	0.67	.0008/.03/.12	74.51
Gupta et al. '18 (SNLI)	33.77	.94/.95/.80	30.24	0.62	**.001/.08/.26**	57.41
Gupta et al. '18 (MSCOCO)	**44.86**	.95/.96/.87	35.87	0.76	.001/.07/.22	63.95
Gupta et al. '18 (Wikipedia)	10.62	.87/.89/.65	16.78	0.65	.001/.06/.22	**55.19**
Gupta et al. '18 (Combined)	37.33	**.96/.97/.89**	**37.21**	0.78	.001/.07/.25	57.66
Seq2Seq (Quora_history)	1.47	.76/.83/.4	4.00	**0.63**	.0006/.02/.07	**60.67**
Seq2Seq (Microsoft_history)	0.01	.66/.62/.41	3.37	0.93	.0005/.005/.02	80.10
Seq2Seq (SNLI_history)	6.13	.84/.87/.54	9.09	0.77	.002/.03/.09	71.57
Seq2Seq (MSCOCO_history)	11.6	**.89/.91/.61**	**16.02**	0.80	.0007/.02/.08	70.52
Seq2Seq (Wikipedia_history)	**15.53**	.85/.89/.51	15.99	0.83	**.006/.07/.15**	76.46
Seq2Seq (Combined_history)	6.04	.88/.9/.57	12.66	0.84	.003/.06/.17	61.01
Gupta et al. '18 (Quora_history)	20.4	.88/.9/.62	18.80	0.75	.005/.10/.24	61.7
Gupta et al. '18 (Microsoft_history)	2.82	.8/.87/.5	9.4	**0.65**	.001/.03/.09	**61.28**
Gupta et al. '18 (SNLI_history)	28.18	.89/.92/.66	21.67	0.81	.004/.09/.22	65.8
Gupta et al. '18 (MSCOCO_history)	23.1	.87/.89/.64	21.35	0.80	**.006/.11/.23**	65.07
Gupta et al. '18 (Wikipedia_history)	22.83	.88/.9/.66	21.90	0.83	.006/.10/.23	65.05
Gupta et al. '18 (Combined_history)	**39.57**	**.94/.94/.81**	**30.6**	0.90	.004/.10/.22	70.92
Ours_{pos} (Quora)	4.45	.83/.88/.55	11.18	**0.47**	.0008/.04/.16	56.33
Ours_{pos} (Microsoft)	1.80	.83/.86/.48	7.80	0.58	.0004/.02/.10	64.32
Ours_{pos} (SNLI)	16.12	.93/.94/.78	28.28	0.58	.001/.08/.27	59.33
Ours_{pos} (MSCOCO)	**40.12**	**.95/.96/.86**	**35.91**	0.78	.001/.06/.22	63.55
Ours_{pos} (Wikipedia)	14.29	.88/.90/.65	17.91	0.65	**.002/.09/.27**	**50.32**
Ours_{pos} (Combined)	22.72	.93/.93/.76	22.75	0.80	.0006/.03/.16	66.17
Ours_{all} (Quora)	7.66	.85/.89/.60	13.06	**0.44**	**.001/.08/.33**	**34.31**
Ours_{all} (Microsoft)	3.06	.85/.87/.50	9.63	0.59	.0008/.03/.15	64.83
Ours_{all} (SNLI)	26.30	.90/.92/.76	25.78	0.56	.001/.06/.26	59.33
Ours_{all} (MSCOCO)	**43.22**	.94/.95/.86	31.06	0.73	.001/.07/.25	55.22
Ours_{all} (Wikipedia)	7.70	.85/.88/.59	12.45	0.63	.001/.06/.23	51.07
Ours_{all} (Combined)	33.02	**.95/.95/.85**	**33.26**	0.84	.001/.05/.17	69.92

Table 2: Examples of generated paraphrases of each model that outputs the highest diversity (mBLEU) score.

Original Sentence: what are some of the best ways to lose 5 pounds in 2 weeks ?
Ground Truth Paraphrase: what are some alternative ways to lose 5 pounds in 2 weeks ?

Model	Generated Paraphrase
Seq2Seq (SNLI)	S1: there is a person in front of a painting of art . S2: there is a person in front of a painting of stairs . S3: there is a person in the picture of a painting
Gupta et al. '18 (Wikipedia)	S1: robinson are some unusual down to receive 5 manufactured in 2 weeks ? S2: robinson are some written period to get 5 manufactured in 2 players ? S3: robinson are some better period to receive 5 manufactured in 3 players ?
Seq2Seq (Quora_history)	S1: what are the good the to face 5 and self the ? S2: what are the of the to face 5 and self the ? S3: what are the of the to get 5 and self the and time ?
Gupta et al. '18 (Microsoft_history)	S1: there are these UNK to UNK operating rates in recent years . S2: there are these UNK to UNK 2.7 losses in recent years . S3: there are these UNK to UNK billion sales in 2 years .
Ours_{pos} (Wikipedia)	S1: what are some big power to achieve 5 feet in 2 weeks ? S2: what are some on power to achieve 5 sauce in 2 weeks ? S3: what are some on power to achieve 5 length in 2 weeks ?
Ours_{all} (Quora)	S1: what are some benefits ways to lose 5 pounds in 2 weeks ? S2: what are some opinion ways to lose 5 pounds in 2 weeks ? S3: what are some alternative ways to lose 5 pounds in 2 weeks ?

Table 3: User study result for a set of 20 original sentences randomly sampled from the test dataset. Each of them underwent inference to produce eight generated paraphrases from the best models described in Section 4.5. The majority of the users have chosen our model, Ours_{all}, which illustrates the highest semantics similarity and diversity among the generated paraphrase sentences from the 20 questions.

Model	Semantic Similarity	Diversity
Seq2Seq	9.09 %	11.81 %
Gupta et al. '18	7.73 %	7.72 %
$\text{Seq2Seq}_{\text{history}}$	15 %	8.63 %
Gupta et al. $\text{'18}_{\text{history}}$	15.45 %	9.55 %
Ours_{pos}	19.54 %	25.45 %
Ours_{all}	**33.18 %**	**36.81 %**

paraphrase, and the first column displays the different types of models used. The second column lists the generated paraphrases. Following previous studies (Gupta et al. 2018), we list only the top three generated paraphrases. We observe that for the generated paraphrases of Seq2Seq (SNLI), Gupta et al. '18 (Wikipedia), Seq2Seq (Quora_history), Gupta et al. '18 (Microsoft_history), and Ours_{pos} (Wikipedia), although they do not contain the same paraphrases within the set, their semantic meanings are different from the original sentence and its reference paraphrase. We further observe that the generated paraphrases from Ours_{all} (Quora) do not only contain different sentences, but their semantic meanings are comparable to that of the original sentence and the ground truth. In addition, we note that the third generated paraphrase has the exact sentence structure as the ground-truth paraphrase.

6 Conclusion

In this study, we analyze the challenges of using style transfer to generate diverse paraphrases, and propose a novel controlled latent vector z', which enables the training of counterfactual debiased instances. We compared our approach to different baselines, including the state-of-the-art, and our experiments show that our proposed approach generates more diverse paraphrases (having similar or higher semantics), both quantitatively and qualitatively.

7 Acknowledgments

This work was supported by IITP grant funded by the Korean government (MSIT) (No. 2017-0-01778, Development of Explainable Human-level Deep machine Learning Framework) and the Creative Industrial Technology Development Program (10053249) funded by the Ministry of Trade, Industry and Energy (MOTIE, Korea).

References

Androutsopoulos, I., and Malakasiotis, P. 2010. A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research*.

Artetxe, M.; Labaka, G.; Agirre, E.; and Cho, K. 2018. Unsupervised neural machine translation. In *ICLR*.

Bowman, S. R.; Angeli, G.; Potts, C.; and Manning, C. D. 2015. A large annotated corpus for learning natural language inference. In *EMNLP*.

Brad, F., and Rebedea, T. 2017. Neural paraphrase generation using transfer learning. In *INLG*.

Chung, J.; Gulcehre, C.; Cho, K.; and Bengio, Y. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

COCO. 2018. Coco common objects in context.

Coster, W., and Kauchak, D. 2011. Simple english wikipedia: A new text simplification task. In *ACL*.

Dai, B., and Lin, D. 2017. Towards diverse and natural image descriptions via a conditional gan. In *ICCV*.

Denkowski, M., and Lavie, A. 2014. Meteor universal: Language specific translation evaluation for any target language. In *EACL Workshop on Statistical Machine Translation*.

Dolan, B.; Brockett, C.; and Quirk, C. 2005. Microsoft research paraphrase corpus. *Retrieved March* 29:2008.

Fan, Z.; Wei, Z.; Li, P.; Lan, Y.; and Huang, X. 2018. A question type driven framework to diversify visual question generation. In *IJCAI*.

Fu, Z.; Tan, X.; Peng, N.; Zhao, D.; and Yan, R. 2017. Style transfer in text: Exploration and evaluation. In *AAAI*.

Goodall, D. W. 1966. A new similarity index based on probability. *Biometrics* 882–907.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *NIPS*.

Gupta, A.; Agarwal, A.; Singh, P.; and Rai, P. 2018. A deep generative framework for paraphrase generation. In *AAAI*.

Guu, K.; Hashimoto, T. B.; Oren, Y.; and Liang, P. 2018. Generating sentences by editing prototypes. In *TACL*.

Huang, S.; Wu, Y.; Wei, F.; and Zhou, M. 2018. Dictionary-guided editing networks for paraphrase generation. *CoRR* abs/1806.08077.

Iyyer, M.; Wieting, J.; Gimpel, K.; and Zettlemoyer, L. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *NAACL*.

Jain, U.; Zhang, Z.; and Schwing, A. G. 2017. Creativity: Generating diverse questions using variational autoencoders. In *CVPR*.

Jhamtani, H.; Gangal, V.; Hovy, E.; and Nyberg, E. 2017. Shakespearizing modern language using copy-enriched sequence-to-sequence models. In *ACL*.

Jiang, Z.; Wen, J.-R.; Dou, Z.; Zhao, W. X.; Nie, J.-Y.; and Yue, M. 2017. Learning to diversify search results via subtopic attention. In *SIGIR*.

Johnson, J.; Alahi, A.; and Fei-Fei, L. 2016. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*.

Kim, H.; Kim, M.; Seo, D.; Kim, J.; Park, H.; Park, S.; Jo, H.; Kim, K.; Yang, Y.; Kim, Y.; Sung, N.; and Ha, J.-W. 2018. Nsml: Meet the mlaas platform with a real-world case study. *arXiv preprint arXiv:1810.09957*.

Kingma, D. P., and Welling, M. 2014. Auto-encoding variational bayes. In *ICLR*.

Lample, G.; Conneau, A.; Denoyer, L.; and Ranzato, M. 2018. Unsupervised machine translation using monolingual corpora only. In *ICLR*.

Li, J.; Galley, M.; Brockett, C.; Gao, J.; and Dolan, B. 2015. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*.

Li, Z.; Jiang, X.; Shang, L.; and Li, H. 2017. Paraphrase generation with deep reinforcement learning. *arXiv preprint arXiv:1711.00279*.

Li, J.; Jia, R.; He, H.; and Liang, P. 2018. Delete, retrieve, generate: A simple approach to sentiment and style transfer. *arXiv preprint arXiv:1804.06437*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Patro, B. N.; Kurmi, V. K.; Kumar, S.; and Namboodiri, V. P. 2018. Learning semantic sentence embeddings using pairwise discriminator. *arXiv preprint arXiv:1806.00807*.

Prabhumoye, S.; Tsvetkov, Y.; Salakhutdinov, R.; and Black, A. W. 2018. Style transfer through back-translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 866–876. Association for Computational Linguistics.

Prakash, A.; Hasan, S. A.; Lee, K.; Datla, V.; Qadir, A.; Liu, J.; and Farri, O. 2016. Neural paraphrase generation with stacked residual lstm networks. *arXiv preprint arXiv:1610.03098*.

Quirk, C.; Brockett, C.; and Dolan, W. 2004. Monolingual machine translation for paraphrase generation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*.

Quora. 2018. First quora dataset release: Question pairs.

Stanford. 2018. The stanford natural language processing group.

Su, J.; Xu, J.; Qiu, X.; and Huang, X. 2018. Incorporating discriminator in sentence generation: a gibbs sampling method. *arXiv preprint arXiv:1802.08970*.

Wieting, J., and Gimpel, K. 2017. Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. *arXiv preprint arXiv:1711.05732*.

Xu, W.; Ritter, A.; Dolan, B.; Grishman, R.; and Cherry, C. 2012. Paraphrasing for style. In *Proceedings of COLING 2012*, 2899–2914. The COLING 2012 Organizing Committee.

Xu, Z.; Liu, B.; Wang, B.; Chengjie, S.; Wang, X.; Wang, Z.; and Qi, C. 2017. Neural response generation via gan with an approximate embedding layer. In *EMNLP*.

Xu, J.; Sun, X.; Ren, X.; Lin, J.; Wei, B.; and Li, W. 2018. Dp-gan: Diversity-promoting generative adversarial network for generating informative and diversified text. *arXiv preprint arXiv:1802.01345*.

Zhou, G.; He, T.; Zhao, J.; and Hu, P. 2015. Learning continuous word embedding with metadata for question retrieval in community question answering. In *ACL*.

Zhu, J.; Park, T.; Isola, P.; and Efros, A. A. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*.