A BIRD'S EYE VIEW ON COHERENCE, AND A WORM'S EYE VIEW ON COHESION

Anonymous authors
Paper under double-blind review

ABSTRACT

Generating coherent and cohesive long-form texts is a challenging problem in natural language generation. Previous works relied on a large amount of human-generated texts to train language models, however, few attempted to explicitly model the desired linguistic properties of natural language text, such as coherence and cohesion. In this work, we train two expert discriminators for coherence and cohesion, respectively, to provide hierarchical feedback for text generation. We also propose a simple variant of policy gradient, called *negative-critical sequence training*, using margin rewards, in which the *baseline* is constructed from randomly generated negative samples. We demonstrate the effectiveness of our approach through empirical studies, showing significant improvements over the strong baseline – attention-based bidirectional MLE-trained neural language model – in a number of automated metrics. The proposed discriminators can serve as baseline architectures to promote further research to better extract, encode, and transfer essential qualities from texts.

1 INTRODUCTION

The terms *coherence* and *cohesion* in linguistics are commonly defined as follows (Williams & Colomb, 1995).

> *Cohesion*: sentence pairs fitting together the way two pieces of a jigsaw puzzle do.
> *Coherence*: what all the sentences in a piece of writing add up to, the way all the pieces in a puzzle add up to the picture on the box.

In layman's terms, *cohesion* indicates that two consecutive sentences are *locally* well-connected, and *coherence* indicates that multiple sentences *globally* hold together.

Generating cohesive and coherent natural language texts that span multiple sentences is a challenging task mainly due to two reasons. First, there is no principled way of modeling cross-sentence linguistic properties, such as cohesion and coherence of a text. Second, there is no widely accepted metric of evaluating the quality of the generated text in terms of cohesion and coherence.

As a result, the task has not been extensively studied in the natural language processing (NLP) community. Most state-of-the-art approaches to natural language generation (NLG) relied on a large amount of human-generated texts to train neural language models (Cho et al., 2014; Sutskever et al., 2014). Although these models can generate sentences that, if judged individually, are similar to human-generated ones, they often fail to capture the local and global dependencies among sentences, resulting in a neither coherent nor cohesive text. For example, neural language models based on Recurrent Neural Networks (RNNs) are widely applied to response generation for dialogue (Vinyals & Le, 2015; Shang et al., 2015; Sordoni et al., 2015; Li et al., 2015). Although the responses by themselves look reasonable, they are either bland such as "I don't know", or incoherent with the whole dialogue session. See Gao et al. (2018) for a comprehensive survey.

In this paper, we strive to address the challenge in a principled manner. We propose a pair of discriminators to score whether and to what extent a text is coherent or cohesive, respectively. The coherence discriminator measures the compatibility among all sentences in a generated text using sentence-level features, thus providing a *bird's eye view* on the text. The cohesion discriminator, on the other hand, measures the compatibility of each pair of consecutive sentences using only word-level features, thus providing a *worm's eye view* on the text. These models, given a conditional input

source sentences	the hotel inglaterra delivered as promised . the staff was welcoming and spoke good english . the cleaning staff did a very good job every day . the rooms were spotless and very modern . the bathroom was large and had a very nice shower , and there were two generously sized bath towels that were twice the size of normal towels .
G_{MLE}	the breakfast in the morning was delicious and very good . it was the only hotel where i slept very well . the staff was very helpful in late afternoon or late times . the breakfast was adequate , with a decent range of cereals , fruit , and fruits . there is also free use of the coffee in the reception area .
$G_{\text{MLE+RL(coherence, cohesion)}}$	the breakfast was plentiful including fresh breads and cooked to order . the location was fantastic . it is in the north of the marina and in a very short distance . the marina has a small swimming pool with sitting area and a small gym . they are very popular and guests have an evening reception which is very nice .

Table 1: Sample generations from our MLE-trained baseline model, G_{MLE}, and our discriminator-guided model $G_{\text{MLE+RL(coherence, cohesion)}}$. The red texts highlight a common problem in G_{MLE} - it exhibits a repetition, and an inconsistent opinion as a review. In contrast, our discriminator-guided model is able to generate a more diverse, and sentiment-consistent continuation.

text and multiple candidate output texts, are learned score the candidate texts with respect to the criterion by optimizing a pairwise ranking loss, respectively. These scores are then used as reward signals to train an RNN-based language model to generate (more) coherent and cohesive texts.

Our main contributions are three-fold: (1) we propose two linguistic discriminators for measuring coherence and cohesion of a text, respectively; (2) we present a simple training mechanism to encode these linguistic properties; and (3) we propose *negative-critical sequence training*, a variant of policy gradient method, which uses negative samples to construct its reward *baseline*.

To the best of our knowledge, this paper is the first attempt to explicitly capture cross-sentence linguistic properties, i.e., coherence and cohesion, for long text generation. Despite the encouraging initial results, we only scratched the surface of the problem. The proposed method is yet to be explored and further improved to meet the ultimate goal of generating meaningful and logical long-form texts. We cast the text generation as an RL problem and review recent papers in Section 2, and we describe our approach in detail in Section 3.

2 Related Work

A word-level sequence generation task can be framed as a reinforcement learning (RL) problem, in which the generator G is viewed as a *policy* π, with parameters θ_π, and each generated word at time t, w_t, is an action to be chosen from a large discrete space, or vocabulary, conditioned on state $s_{t-1} = w_{\leq t-1}$.

Let r_t be the reward for a partially generated text sequence $w_{\leq t}$. We define the long-term expected reward $\mathcal{J}(\pi) = \mathbb{E}_{s_0 \sim q, \pi}[\sum_{t=1}^{\infty} \gamma^{t-1} r_t]$, where q is the initial distribution of conditional input texts. From Sutton et al. (1999), the gradient of $\mathcal{J}$ with respect to θ_π is

$$\nabla_{\theta_\pi} \mathcal{J} = \mathbb{E}_{s \sim \rho^\pi, a \sim \pi(\cdot|s)}[Q^\pi(s,a) \nabla_{\theta_\pi} \log \pi_{\theta_\pi}(a|s)] \tag{1}$$

where ρ^π is the stationary distribution and $Q^\pi(s,a)$ is the expected return from state s and taking action a, both following policy π. For brevity, we omit the technical derivation. In our work, we treat the text generation task as episodic with length L, only end-of-episode rewards r_L and $\gamma = 1$.

There are many works that tune neural language models via reward signals, such as Ranzato et al. (2015) and Paulus et al. (2017). These works optimize for specific NLP metrics, such as BLEU (Papineni et al., 2002) or ROUGE (Lin & Hovy, 2003), using REINFORCE (Williams, 1992; Sutton et al., 1999). However, it is well-known that these metrics do not give a complete picture on the generation quality. Only recently has there been efforts to provide more relevant, trainable measures for which to optimize (Li et al., 2015; Holtzman et al., 2018) concerning repetition, and fit between text pairs, but these works use the measures to re-rank outputs, not to reward or penalize outputs. Li et al. (2016) constructed reward models, such as information flow and semantic coherence, to tune the generator, yet they do not provide an ablation study on individual effects of their reward models.

Another line of research is to use Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) in order to incorporate feedback signals for text generation (Yu et al., 2017; Lin et al., 2017; Zhang et al., 2017c; Guo et al., 2017; Fedus et al., 2018; Zhang et al., 2018). However, the discriminator in these works are trained to distinguish real texts from the generated ones, operating as a black-box rather than providing feedback on particular linguistic aspects of the texts. In fact, Yang

et al. (2018) has partially addressed this issue by training the discriminator as a language model. Although the discriminator provides a fine-grained feedback at the word-level, it does not critique on different linguistic properties of generated samples.

These generator models, when faced with a long-form generation task that span multiple sentences, are far from being perfect and exhibit some critical errors, such as a breakdown of local connections between consecutive sentences (cohesion), let alone globally solid intent (coherence). As a result, readers can easily take these cues and discriminate against such generations. We argue that the primary reason comes from a lack of injecting some key qualities in the generation, which motivates our work.

3 MODEL

We assume that global soundness is governed by how individual sentences with different meanings are organized to form a single paragraph. So we restrict our evaluation of coherence solely on the sentence-level. If the meanings of sentences are not organized in a correct manner, we have difficulty in picking up the message of the paragraph as a whole, regardless of seamless local connectivity between consecutive sentences.

This is not to say that a local connection between any two sentences should be overlooked. One can easily distinguish a model-generated sentence from a real sentence, simply by looking at whether an arbitrary sentence is followed by a logically sound sentence, not to speak of grammar.

We instill these two different, yet important concepts in two discriminators, respectively. Our two discriminators play a distinct role, each independently operating on the sentence-level and on the word-level. Our models closely resemble successful works in the computer vision domain, such as StackGAN (Zhang et al., 2017a;b) and PatchGAN (Isola et al., 2017) since our models provide hierarchical signals that are derived from the same raw low-level data to the generator. We call the sentence-level discriminator as the *coherence* discriminator $D_{\text{coherence}}$, and the word-level discriminator as the *cohesion* discriminator D_{cohesion}.

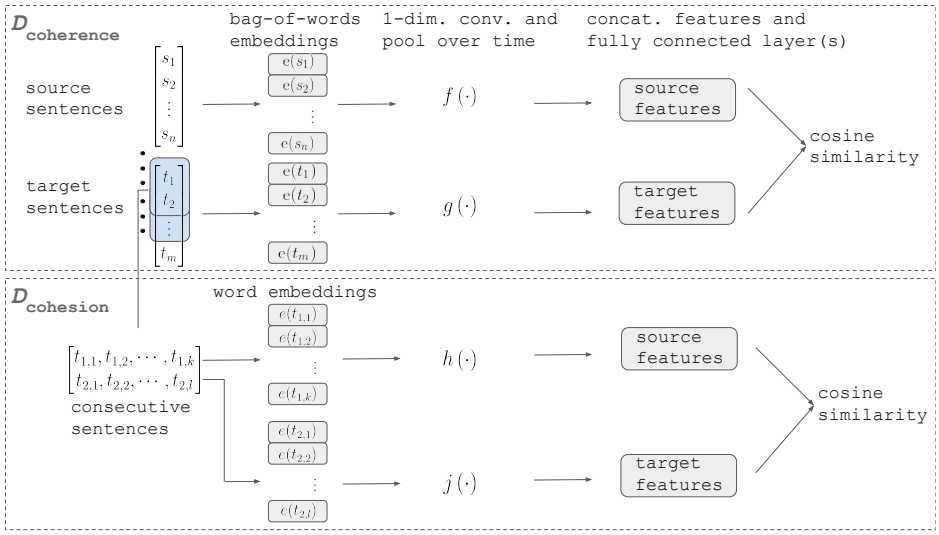

Figure 1: Illustration of coherence and cohesion discriminators. $D_{\text{coherence}}$ takes in bag-of-words sentence embeddings as inputs, and D_{cohesion} takes in the raw word embeddings of consecutive sentences as inputs.

3.1 COHERENCE DISCRIMINATOR: $D_{\text{COHERENCE}}$

Given n source input sentences $S := [s_1, s_2, ..., s_n]$, m target output sentences $T := [t_1, t_2, ..., t_m]$, and $\widetilde{m}$ generated sentences $\widetilde{T} := [\widetilde{t}_1, \widetilde{t}_2, ..., \widetilde{t}_{\widetilde{m}}]$, this discriminator's task is to distinguish a real pair (S, T) from a synthetic pair $(S, \widetilde{T})$.

Design. Our design of $D_{\text{coherence}}$ is inspired by the Deep Semantic Similarity Model (DSSM) (Huang et al., 2013; Gao et al., 2014; Xu et al., 2017). Given source input sentences S and target output sentences T, $D_{\text{coherence}}$ independently processes S and T through different convolutional networks[1], followed by a cosine similarity function. Since the final layer of the discriminator is a cosine similarity function, the scale of coherence is between -1 and 1, where 1 indicates the maximal coherence score, and -1 the minimal coherence score. See Figure 1 for an illustration.

$D_{\text{coherence}}$ is designed to determine whether S and T together add up to a single coherent message. In particular, $D_{\text{coherence}}$ learns to inspect the higher-level role of T, such as but not limited to, whether it supports the intent of S, transitions smoothly against S, or avoids redundancy. We encode these features with a hinge-based triplet ranking loss (Wang et al., 2014; Kiros et al., 2014). Furthermore, instead of learning to rank between a pair of target texts given a source text, the discriminator learns to correctly rank against multiple target texts, in particular, against the hardest negative target text, which is in the same vein as Faghri et al. (2017). This eventually boils down to a retrieval problem given a query, i.e. searching for the correct target text (item) given the source text (query).

Therefore, to evaluate $D_{\text{coherence}}$, we employ the metrics commonly used in information retrieval, such as recall at *K* (R@*K*), which is defined as the fraction of correctly identifying an item in the TOP-*K* retrieved list (Baeza-Yates et al., 1999). We present the retrieval results on test data in Table 2.

Discriminators	**Target Sentences Retrieval**		
	R@1	**R@5**	**R@10**
$D_{\text{coherence}}$	0.17	0.40	0.60
D_{cohesion}	0.10	0.28	0.43

Table 2: Retrieval ratios for coherence and cohesion discriminators from a collection of 100 negative candidates. The reported numbers are averages over 20 evaluations.

Training mechanism. How do we construct these list of candidate target sentences T, given the source sentences S? We assume the T that follows an S in the data is a positive target sample, or the correct item to retrieve. Negative samples are constructed using three different methods within a batch while iterating through the training data, motivated by Wieting et al. (2016):

- The first method is to simply rotate T with S fixed in a batch. For a single S, this method yields $B-1$ negative samples, where B is the batch size.
- The second method is to shuffle the sentence order once, different from its original order, known as a derangement, in each T to break coherence, and this yields one negative sample.
- Lastly, we combine the previous two methods: rotate T across a batch and shuffle sentences within T, yielding $B-1$ negative samples.

These $2B-1$ negative samples and a single positive sample, in sum, pose a significant challenge in learning. To fit this training task into a ranking framework, we optimize over $D_{\text{coherence}}(S,T) - \max_{i\in\{1,\ldots,2B-1\}} D_{\text{coherence}}(S,\widetilde{T}_i)$, in which we take the most challenging negative sample.

For this discriminator, we limit inputs to sentence embedding vectors but no word embedding vectors. In other words, we feed in single sample $[s_1, s_2, ..., s_n]$ and $[t_1, t_2, ..., t_m]$ in sentences instead of words that comprise the sentences. In this manner, we can strictly delineate $D_{\text{coherence}}$ on the sentence-level, rather than allowing finer judgments on the word-level. This pushes the discriminator to focus on higher-level information that directly affect coherence.

Then the natural question is, how do we obtain a general-purpose or *near* ground truth sentence embedding vector for each of the real source sentences $[s_1, s_2, ..., s_n]$ and the target sentences $[t_1, t_2, ..., t_m]$? We simply take an average of the pre-trained word embedding vectors of each sentence as its sentence-level embedding. Empirically, this averaging scheme was shown to be highly effective on sentence similarity and entailment tasks (Wieting et al., 2016; Arora et al., 2017).

[1]We explored with deeper networks. However, the performance difference was marginal. For simplicity, we decided to use a 1-layer convolutional network architecture (Kim, 2014; Collobert et al., 2011).

3.2 COHESION DISCRIMINATOR: D_{COHESION}

Our second discriminator pays attention only to low-level features, such as grammar of each of the sentences and the logical flow between arbitrary two consecutive sentences. These two aspects combined, despite more linguistic aspects that we do not mention, heavily influence readability.

For simplicity, D_{cohesion} is similar to $D_{\text{coherence}}$, except its architecture, input, and negative sample construction are modified to encode cohesion between any pair of sentences on the word-level. A single input sample to D_{cohesion} is a pair of two consecutive sentences: $[s_{i,1}, s_{i,2}, ..., s_{i,n}]$ and $[s_{i+1,1}, s_{i+1,2}, ..., s_{i+1,m}]$, where $s_{i,k}$ denotes the k-th word in sentence i. We construct the negative samples by following the three ways mentioned above, where shuffling occurs on word level within each sentence, rather than shuffling multiple sentences on the sentence level.

4 NEGATIVE-CRITICAL SEQUENCE TRAINING

Actor-critic methods (Witten, 1977; Barto et al., 1983) parametrized by neural networks typically require learning a separate critic network. In NLP, we have seen similar practices by Ranzato et al. (2015), Bahdanau et al. (2016), and Nguyen et al. (2017). Recently, however, Rennie et al. (2017) proposed an effective self-critical sequence training (SCST) mechanism that avoids learning a new critic network. Similarly, our method does not require learning a separate critic network, rather we use discriminator scores of negative samples to create a *baseline*.

For an arbitrary pair of S and T_{gen}, which is the generator's output conditioned on S, we compute the coherence and cohesion scores by calling $D_{\text{coherence}}$ and D_{cohesion}, respectively. Since each review consists of multiple sentences, the overall cohesion score is computed as the average of scores of all consecutive sentence pairs. These scalar scores, however, have no interpretation since the discriminators are trained by optimizing a margin ranking loss. Instead, the differences between positive sample scores and the maximal or average negative sample scores provide insight of how well the models can distinguish between the positives and the negatives. Therefore, these margins can be considered as rewards with baselines, and thus we define the reward functions as:

$$R_{\text{coherence}}(S, T) \coloneqq D_{\text{coherence}}(S, T) - \mathbb{E}_{\widetilde{T}}[D_{\text{coherence}}(S, \widetilde{T})]$$
$$R_{\text{cohesion}}(s_i, s_{i+1}) \coloneqq D_{\text{cohesion}}(s_i, s_{i+1}) - \mathbb{E}_{\widetilde{s}_{i+1}}[D_{\text{cohesion}}(s_i, \widetilde{s}_{i+1})]$$

where $\tilde{\cdot}$ denotes a negative sample for a given source condition, and $\mathbb{E}_{\widetilde{T}}$ (and $\mathbb{E}_{\widetilde{s}_{i+1}}$) are computed by averaging over an ensemble of negative samples. Notice that this reward resembles the ranking loss we use to train our discriminators, except that our baseline is an average score (instead of the maximal score) over negative samples. The rational for this difference is that: the maximal score baseline is in fact noisy to be used as rewards, because the best randomly constructed negative samples may be a *formidably* good sample. To alleviate such noise, we use the average discriminator scores of negative samples as the *baseline*, and this turns out to be an empirically better alternative.

Finally, we use policy gradient (Williams, 1992; Sutton et al., 1999) to maximize a weighted combination of the coherence and cohesion rewards. For illustrative purposes, we equally weigh them for updating our policy, i.e., the generator G.

5 EXPERIMENTS

In this section, we show results of training both $D_{\text{coherence}}$ and D_{cohesion}, and compare our RL-tuned generators $G_{\text{MLE+RL(cohesion)}}$, $G_{\text{MLE+RL(coherence)}}$, and $G_{\text{MLE+RL(coherence, cohesion)}}$ with the baseline model G_{MLE}. We argue that through the use of feedback from our simple discriminators to G_{MLE}, the quality of text generations improves significantly. See Table 1 for a comparison.

Dataset. We use the publicly available TripAdvisor's hotel reviews dataset collected by Wang et al. (2010). We consider only a subset of the reviews dataset satisfying the following two conditions: a review must have (1) at least 10 sentences, and (2) each sentence should have more than 5 and less than 30 words. This yields roughly 60,000 reviews, split into $[0.8, 0.1, 0.1]$ ratio for train/dev/test. We merge the source and target vocabularies, and limit it to the top 50,000 frequent words, excluding special tokens. For each of these reviews, as in Holtzman et al. (2018), we consider the first five sentences as the source input S to G, and the following five sentences as the target output T from G.

Evaluation metrics. It is widely known that there is no accurate metric to evaluate the generator. Nevertheless, we report scores of standard metrics, such as negative log-likelihood (NLL), perplexity (PPL), BLEU and proportion of unique n-grams within a single generation (intra-unique-n), and across generations (inter-unique-n), as in Gu et al. (2018). Results are shown in Table 3.

Model	NLL	PPL	BLEU-3	BLEU-4	BLEU-5	intra-unique-1	intra-unique-2	inter-unique-1	inter-unique-2	inter-unique-3	length ratio
G_{MLE} (baseline)	0.37	1.45	0.51	0.31	0.16	0.68	0.95	0.02	0.16	0.35	1.01
$G_{\text{MLE+RL(cohesion)}}$	**0.36**	**1.44**	**0.54**	**0.33**	**0.18**	0.68	0.95	0.01	0.10	0.23	0.99
$G_{\text{MLE+RL(coherence)}}$	0.37	1.44	0.53	0.32	0.16	**0.69**	**0.96**	**0.07**	**0.39**	**0.72**	0.97
$G_{\text{MLE+RL(coherence, cohesion)}}$	0.37	1.45	0.52	0.31	0.16	0.69	0.96	0.02	0.18	0.40	0.96

Table 3: An ablation study with automated evaluation metric scores: NLL, PPL, BLEU-n, intra/inter-unique-n, along with the length ratio with the length of corresponding true target sentences as 1. Results show that our proposed discriminators improve, albeit marginally, in BLEU scores and significantly in diversity, an escape from generating *generic* responses, or mode collapse, without compromising on NLL or PPL. We used equally weighted rewards, and numbers are highlighted in **bold** before rounding.

5.1 Implementation details

G takes individual words as inputs and embeds into a pre-trained 300-dimensional word vectors from GloVe (Pennington et al., 2014). This embedding layer is fixed throughout training. G uses a gated recurrent unit with two layers and a hidden size of 1024 for both bidirectional encoder and attention-based decoder. During optimization using Adam (Kingma & Ba, 2014), we set the learning rate to 2e-4 and clip the gradient's L2-norm to 1.0. We initially train G_{MLE} by maximizing the word-level likelihood estimation (MLE) from data that consist of positive samples, for 50 epochs. This is our baseline model against which to empirically prove value of our hierarchical discriminators.

$D_{\text{coherence}}$ also uses the pre-trained GloVe word vectors[2], which are fixed. The source processing network and the target processing network have the same structure, but different parameters. The convolutional layer has filters of sizes 2, 3, 4, and 5, each with 256 filters. Each convolution filter is followed by a batch normalization layer and $\tanh$ activation. Then we max-pool in time over the features and append two fully connected layers (1024, 2048) with ReLU and $\tanh$ activation, respectively. We use an Adam optimizer with a learning rate of 1e-5.

D_{cohesion} is similar to $D_{\text{coherence}}$. It has convolutional filters of sizes 3, 4, 5, and 6, each with 256 filters, followed by $\tanh$ activation, a batch normalization layer over max-pooled features, and one fully-connected layer (2048) with $\tanh$ activation. We train both discriminators for 100 epochs and choose models with the best R@1 validation scores.

In the tuning stage, we use the negative-critical sequence training as explained in Section 4 up to 5 epochs, with a learning rate of 1e-5. We also continue with supervised learning to G to limit the policy search within a grammatically correct space, similar to Paulus et al. (2017); Wu et al. (2016); Lewis et al. (2017). In practice, sequence-level rewards are only available upon a completed generation, so they are sparse signals for the generator. Typically, sparse end-of-sequence rewards entail a noisy training, yet would want the learning generalize to the testing data. We observed that, for our particular task, most noises were caused by exploration, and the learning generalized to the testing data. Thus, reward shaping was unnecessary, unlike previous works (Li et al., 2017; Yang et al., 2018) that further provided signals for partially generated sequences.

5.2 Sanity check on $D_{\text{coherence}}$ and D_{cohesion}

Most of the reviews written by the hotel guests are considered coherent. Suppose we randomly select a negative sample from a pool of other continuations $\widetilde{T}$ in the data. Even for a layman in linguistics, one can effortlessly discern if the review repeats similar ideas, albeit in different wordings, whether $\widetilde{T}$ supports or contradicts S and is considered as a natural continuation from S. To show that our $D_{\text{coherence}}$ does these jobs and likewise for D_{cohesion}, we show some randomly selected positive and negative samples and their assigned margin scores in Table 4.

[2]The vector dimension can be different from that of G. The differences were marginal for sizes 50, 100, and 300. For results shown in this paper, we used the same dimension of size 300.

source	**cohesion**	**coherence**
this hotel was unbelievably overpriced .	0.0002	
we were looking for something cheaper but thought we would at least be staying in a decent hotel having paid that much when booking .	0.0411	
it wasn t clear when booking that we would have to share a bathroom .	0.0084	
there was one shower for the whole floor which was tiny and unclean .	0.0054	
the room was old and lacking in facilities .		
target		
the beds were very uncomfortable and the linen was very old .	0.0768	
breakfast was ok , but the staff were incompetent .	0.0591	
on our last day they were too lazy to clean our table and never bothered taking our order .	-0.0097	**+0.3735**
we had to leave having had no breakfast , as we ran out of time .	0.0457	
they saw us get up and leave and didn t even apologise for the appalling lack of service .		
negative target		
the staff recommended great restaurants with very reasonable prices within walking distance .	0.0514	
the paris hop on bus stops nearby .	0.0798	
the gare l est is within 3 blocks .	-0.0156	
we paid 75 euro per nite excluding breakfast but paid for breakfast one day and found it very good and reasonably priced .	0.0082	**-0.2001**
the rooms are clean and bathrooms ensuite .		
more examples of cohesion		
once you get there you are greeted by the staff . they explain everything to you , and in english , not the best , but good enough .	0.1004	
the coffee was even good for a coffee snob like myself . the hotel is much smaller than i thought and only has six floors .	-0.1103	
the only negative was the curtain in the bathroom . it was very shear and we felt that people in the building across the street could look right in at night .	0.0787	
the beer at the lobby bar was stale . there are many friendly cats on the grounds .	-0.0830	

Table 4: Coherence and cohesion margin scores on test data. The cohesion score at the end of each line is computed with its next sentence. This is a common example of contradiction and inconsistent sentiment, implying incoherence. We append more examples with extreme cohesion margin scores.

6 Discussion

We first comment on the text-to-text retrieval results in Table 2. Compared to image-to-textual caption retrieval tasks, the numbers are lower. One plausible reason is that an image is rich in semantics: *an image is worth a thousand words*. In contrast, a few sentences are limited in their capacity to convey a message, in addition to grammar and readability constraints. For this reason, the discriminator models face a more difficult challenge in identifying the correct query (source sentences) - item (target sentences) pairs.

Furthermore, we note that our methods to construct negative samples, in spite of its simplicity, are not thorough. For example, a randomly selected next sentence, given an arbitrary sentence, may actually be a valid continuation for D_{cohesion}. In this work, given an unlabelled dataset, we ignore such a problem, which may not be negligible and explain why we see a lower performance compared to that of $D_{\text{coherence}}$. Despite potential drawbacks of our methodology, we have shown significant improvements with imperfect discriminators. After experimenting with different architectures and hyper-parameters, we conclude that the table numbers are reasonable for the task.

We do note that results will get better with more data - our discriminators, as well as the generator, will be well-trained by seeing more data samples. We consider the dataset to be rather small because each pre-processing condition is quite restrictive. However, our goal is to demonstrate the efficacy of our discriminator models, rather than to show good results arising from a large amount of data.

6.1 What do $D_{\text{coherence}}$ and D_{cohesion} capture?

Ideally, given how $D_{\text{coherence}}$ is constructed: *independently* processes the source and target sentences, we would want $D_{\text{coherence}}$ to detect whether the target sentences support the message delivered in the source sentences. Some cues that signal lacking such support are repetitive, contradicting, irrelevant,

or sentiment inconsistent statements without appropriate transition phrases. Although $D_{\text{coherence}}$ only computes scores and does not reason out with specific cues, we observed that it learns to pick up these ideas that govern coherence of a multi-sentence paragraph. Despite some randomness from scores of randomly constructed negative samples, its critique margin scores are fairly consistent, based on a collection of margin scores on hotel reviews. Examples of more detailed incoherent aspects $D_{\text{coherence}}$ learns to distinguish include parts of a review commenting on different countries, cities or locations, and listing prices in different currency denominations. This is a positive result of training on randomly selected $\widetilde{T}$ in the same batch.

The role of D_{cohesion} is similar to $D_{\text{coherence}}$, except it operates in-between sentences. We observed that D_{cohesion} captures some low-level logical connection. For example, it favorably scores artifacts that are commonly mentioned together from positive samples, and this is again a result of our methods to construct negative samples through shuffling and rotating target sentences in the same batch. As a consequence, artifacts that do not appear together as many times in consecutive sentences yield low cohesion scores.

However, we note that low cohesion scores among consecutive sentences do not necessarily imply a bad writing. Although one needs to avoid consistently low cohesive writing, the writer may simply enumerate seemingly disparate aspects written into respective sentences, and these likely imply a low connection between immediately neighboring sentences. Therefore, we do not solely optimize for the cohesion criterion.

6.2 Potential improvements in our approach

We would also like to comment on our *imperfect* discriminators. When we tuned G for up to as many epochs as in the pre-training stage, we noticed that G learns to find a policy that maximizes the discriminators' margin scores, yet diverge from what we consider a good writing. Although this problem can be partially overcome by training on a larger dataset, we determined that these two discriminators do not give a comprehensive critique, and there exist other equally, if not more, important linguistic properties that we did not address. We hope to extend our hands-free model framework to encode these features and provide richer signals for G to improve upon.

While reinforcing linguistic properties such as coherence and cohesion is the first attempt and an important research direction, we consider our results to be preliminary, and many of our experiment figures allude to plenty of room for further improvements, such as recall scores. We admit that we can argue the structure of $D_{\text{coherence}}$ and D_{cohesion} either way, whether to process the source and target sentence(s) independently with different parameters, or together. Nevertheless, we are convinced that our architecture for both $D_{\text{coherence}}$ and D_{cohesion} generalize well to unseen texts, and we plan to provide online a collection of examples with corresponding margin rewards, due to limited space in this paper, and release all of our resources to promote this line of research.

7 Conclusion

In this paper, we proposed to encode essential linguistic properties, in particular, coherence and cohesion, with a simple neural network architecture, and quantify into negative-critical margin scores. The coherence discriminator, $D_{\text{coherence}}$, assesses the compatibility between source sentences and target sentences as a single paragraph, with sentence-level inputs but no specific word-level inputs, thus a bird's eye view on coherence. On the other hand, the cohesion discriminator, D_{cohesion}, focuses on low-level features using raw word-level inputs, and assesses the compatibility between all consecutive sentences.

The scores computed by these discriminators are used as reward signals in our proposed policy gradient, and proved to be effective in tuning the neural language models. Based on automated NLP metrics, we showed a significant improvement, specifically in escaping mode collapse while maintaining readability, which has been a problem when using the sequence-to-sequence framework.

Future work will focus on casting the long-form text generation task into the GANs framework with coherence and cohesion discriminators learning against the neural language model, and providing tailored signals. It is an extension of GANs to multiple discriminators, similar to Durugkar et al. (2016), but each discriminator enforces a distinct linguistic property in shaping the generator.

REFERENCES

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A simple but tough-to-beat baseline for sentence embeddings. 2017.

Ricardo Baeza-Yates, Berthier Ribeiro-Neto, et al. *Modern information retrieval*, volume 463. 1999.

Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron Courville, and Yoshua Bengio. An actor-critic algorithm for sequence prediction. *arXiv preprint arXiv:1607.07086*, 2016.

Andrew G Barto, Richard S Sutton, and Charles W Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE transactions on systems, man, and cybernetics*, (5):834–846, 1983.

Kyunghyun Cho, Bart van Merriënboer, Çalar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, Doha, Qatar, October 2014. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/D14-1179`.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537, November 2011. ISSN 1532-4435. URL `http://dl.acm.org/citation.cfm?id=1953048.2078186`.

Ishan P. Durugkar, Ian Gemp, and Sridhar Mahadevan. Generative multi-adversarial networks. *CoRR*, abs/1611.01673, 2016. URL `http://arxiv.org/abs/1611.01673`.

Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: improved visual-semantic embeddings. *arXiv preprint arXiv:1707.05612*, 2017.

William Fedus, Ian Goodfellow, and Andrew Dai. Maskgan: Better text generation via filling in the ____. 2018. URL `https://openreview.net/pdf?id=ByOExmWAb`.

Jianfeng Gao, Patrick Pantel, Michael Gamon, Xiaodong He, and Li Deng. Modeling interestingness with deep neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2–13, 2014.

Jianfeng Gao, Michel Galley, and Lihong Li. Neural approaches to conversational ai. *arXiv preprint arXiv:1809.08267*, 2018.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 27*, pp. 2672–2680. Curran Associates, Inc., 2014. URL `http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf`.

Xiaodong Gu, Kyunghyun Cho, JungWoo Ha, and Sunghun Kim. Dialogwae: Multimodal response generation with conditional wasserstein auto-encoder. *CoRR*, abs/1805.12352, 2018. URL `http://arxiv.org/abs/1805.12352`.

Jiaxian Guo, Sidi Lu, Han Cai, Weinan Zhang, Yong Yu, and Jun Wang. Long text generation via adversarial training with leaked information. *arXiv preprint arXiv:1709.08624*, 2017.

Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. Learning to write with cooperative discriminators. In *Proceedings of the Association for Computational Linguistics*, 2018.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry P. Heck. Learning deep structured semantic models for web search using clickthrough data. In *CIKM*, 2013.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017.

Yoon Kim. Convolutional neural networks for sentence classification. In *EMNLP*, 2014.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

Ryan Kiros, Ruslan Salakhutdinov, and Richard S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *CoRR*, abs/1411.2539, 2014. URL http://dblp.uni-trier.de/db/journals/corr/corr1411.html#KirosSZ14.

Mike Lewis, Denis Yarats, Yann N Dauphin, Devi Parikh, and Dhruv Batra. Deal or no deal? end-to-end learning for negotiation dialogues. *arXiv preprint arXiv:1706.05125*, 2017.

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

Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation. *arXiv preprint arXiv:1606.01541*, 2016.

Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. Adversarial learning for neural dialogue generation. *arXiv preprint arXiv:1701.06547*, 2017.

Chin-Yew Lin and Eduard Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pp. 71–78, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics. doi: 10.3115/1073445.1073465. URL https://doi.org/10.3115/1073445.1073465.

Kevin Lin, Dianqi Li, Xiaodong He, Zhengyou Zhang, and Ming-Ting Sun. Adversarial ranking for language generation. In *Advances in Neural Information Processing Systems*, pp. 3155–3165, 2017.

Khanh Nguyen, Hal Daumé, and Jordan L. Boyd-Graber. Reinforcement learning for bandit neural machine translation with simulated human feedback. In *EMNLP*, 2017.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pp. 311–318, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://doi.org/10.3115/1073083.1073135.

Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. *CoRR*, abs/1705.04304, 2017. URL http://arxiv.org/abs/1705.04304.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014. URL http://www.aclweb.org/anthology/D14-1162.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. *CoRR*, abs/1511.06732, 2015. URL http://arxiv.org/abs/1511.06732.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1179–1195, 2017.

Lifeng Shang, Zhengdong Lu, and Hang Li. Neural responding machine for short-text conversation. *arXiv preprint arXiv:1503.02364*, 2015.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. A neural network approach to context-sensitive generation of conversational responses. In *NAACL-HLT*, May 2015.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 27*, pp. 3104–3112. Curran Associates, Inc., 2014. URL http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf.

Richard S. Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Proceedings of the 12th International Conference on Neural Information Processing Systems*, NIPS'99, pp. 1057–1063, Cambridge, MA, USA, 1999. MIT Press. URL http://dl.acm.org/citation.cfm?id=3009657.3009806.

Oriol Vinyals and Quoc Le. A neural conversational model. *ICML Deep Learning Workshop*, 2015.

Hongning Wang, Yue Lu, and ChengXiang Zhai. Latent aspect rating analysis on review text data: a rating regression approach. In *KDD*, 2010.

Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1386–1393, 2014.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. Towards universal paraphrastic sentence embeddings. *ICLR*, 2016.

J.M. Williams and G.G. Colomb. *Style: Toward Clarity and Grace*. Chicago guides to writing, editing, and publishing. University of Chicago Press, 1995. ISBN 9780226899152. URL https://books.google.com/books?id=wheCdBUIpwIC.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3-4):229–256, May 1992. ISSN 0885-6125. doi: 10.1007/BF00992696. URL https://doi.org/10.1007/BF00992696.

Ian H Witten. An adaptive optimal controller for discrete-time markov environments. *Information and control*, 34(4):286–295, 1977.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. *arXiv preprint*, 2017.

Zichao Yang, Zhiting Hu, Chris Dyer, Eric P Xing, and Taylor Berg-Kirkpatrick. Unsupervised text style transfer using language models as discriminators. *arXiv preprint arXiv:1805.11749*, 2018.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *AAAI*, 2017.

Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017a.

Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N. Metaxas. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *CoRR*, abs/1710.10916, 2017b. URL http://arxiv.org/abs/1710.10916.

Yizhe Zhang, Zhe Gan, Kai Fan, Zhi Chen, Ricardo Henao, Dinghan Shen, and Lawrence Carin. Adversarial feature matching for text generation. In *NIPS*, 2017c.

Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiujun Li, Chris Brockett, and Bill Dolan. Generating informative and diverse conversational responses via adversarial information maximization. In *NIPS*, 2018.