Improving Adversarial Text Generation with n-Gram Matching

Shijie Li
Faculty of Engineering and IT
University of Technology Sydney
Australia
Shijie.Li@student.uts.edu.au

Massimo Piccardi
Faculty of Engineering and IT
University of Technology Sydney
Australia
Massimo.Piccardi@uts.edu.au

Abstract

In the past few years, generative adversarial networks (GANs) have become increasingly important in natural language generation. However, their performance seems to still have a significant margin for improvement. For this reason, in this paper we propose a new adversarial training method that tackles some of the limitations of GAN training in unconditioned generation tasks. In addition to the commonly used reward signal from the discriminator, our approach leverages another reward signal which is based on the occurrence of n-gram matches between the generated sentences and the training corpus. Thanks to the inherent correlation of this reward signal with the commonly used evaluation metrics such as BLEU, our approach implicitly bridges the gap between the objectives used during training and inference. To circumvent the non-differentiability issues associated with a discrete objective, our approach leverages the reinforcement learning policy gradient theorem. Our experimental results show that the model trained with mixed rewards from both n-gram matching and the discriminator has been able to outperform other GAN-based models in terms of BLEU score and quality-diversity trade-off at a parity of computational budget.

1 Introduction

Neural language generation (NLG) is an important research area in natural language processing (NLP) because of its fundamental role in many other tasks such as machine translation (Wu et al., 2016), text summarization (See et al., 2017), and dialog systems (Bordes and Weston, 2017). Typically, these models are trained using an approach called "teacher forcing", where the probability of ground-truth tokens conditional on previous ground-truth tokens is maximized during training (Goyal et al., 2016). Although this approach has reported substantial success, it biases the model toward ground-truth samples during training, but it has to rely on its own generated samples during inference, often resulting in a mismatch between the two data distributions and the so-called "exposure bias" problem. Researchers have made several attempts to alleviate this issue, including scheduled sampling (Bengio et al., 2015), adversarial training (Yu et al., 2017), and, more recently, optimal transport objectives (Wang et al., 2020).

Generative adversarial networks (GANs) (Goodfellow et al., 2014), initially conceived for realistic image generation, have attained a good degree of success also in a variety of NLP tasks. Yu et al. (2017) has been one of the earliest attempts to demonstrate the potential of GANs for NLG. Following that, a series of related works have shown considerable improvements in terms of sample quality. However, some of these approaches rely on sophisticated architecture design such as LeakGAN (Guo et al., 2018), while others like RelGAN (Nie et al., 2019) are based on enhanced model capacities. Therefore, direct comparison of the performance between different models becomes difficult since the quality of the generated sentences is directly related to the model's capacity (Radford et al., 2019). For this reason, in this paper we adopt a unified benchmark called Texygen (Zhu et al., 2018; Lu et al., 2018), which re-implemented the models of several significant works, but at an

approximate parity of capacity to permit insightful comparisons.

Unlike other NLG tasks such as machine translation or text summarization, unconditioned generation refers to the generation of natural sentences without any input or prompt during training and specific ground-truth target sentences during evaluation (Semeniuta et al., 2018). It is an important task to test the model's ability to generate realistic and diverse sentences. Therefore, calculating performance based on sentence-level metrics becomes impossible (Lin, 2004; Lavie and Agarwal, 2007). As a consequence, researchers have had to adopt corpus-level metrics such as corpus BLEU (Papineni et al., 2002) and self-BLEU (Zhu et al., 2018), and utilize the entire reference set to evaluate each generated sentence (Zhu et al., 2018; Lu et al., 2018). Despite some controversy, GAN-based models have principled advantages in unconditioned generation tasks: by leveraging a discriminator trained over the entire training corpus, the generator can be effectively rewarded or penalized. On the other hand, GAN-based models also face shortcomings, the most obvious of which is that the reward signal from the discriminator can prove elusive to interpret (Lu et al., 2018).

To tackle the above limitations and retain the advantages of GANs in unconditioned language generation, in this paper we propose a novel reward to be used in adversarial training. The proposed reward leverages the matching information between the generated sentences and the entire training corpus, and we therefore aptly refer to our approach as n-gram **M**atching-**E**nhanced GAN (n-MEGAN). Since n-gram matching strongly correlates with the BLEU score used for evaluation, our approach manages to alleviate the mismatch between training and inference. At the same time, since it does not directly optimize for the BLEU score during training, it is able to prevent overfitting and improve performance compared to its baseline and a range of competitive, comparable models.

2 Related Work

While the original GAN used a differentiable training objective, later GAN-based methods have been able to also incorporate non-differentiable components, and can be roughly divided into two categories based on how they address non-differentiability (Semeniuta et al., 2018). The first category aims to ensure end-to-end trainability through continuous relaxations, such as the Gumbel-Softmax relaxation used in Kusner (2016) and Nie (2019). The other category circumvents this issue by drawing on reward-based algorithms, especially the policy gradient theorem (Williams, 2004) of reinforcement learning (RL). One advantage of this approach is that it allows for the direct optimization of the discrete metrics which are extensively used for the evaluation of NLG tasks. The most straightforward candidate for this is the BLEU score (Ranzato et al., 2016). However, training the model with the same metric used for its evaluation is prone to overfitting, and using BLEU as reward has not led to reported improvements (Casas et al., 2018; Lu et al., 2018).

In view of this, the majority of GAN-based works in unconditioned language generation has continued to explore differentiable objectives. Guo et al. (2018) have proposed a complex hierarchical structure that allows the reward signal from the discriminator to better flow into the generator, and showed its advantages in long text generation. Fedus et al. (2018) have focused on integrating text understanding into text generation by randomly masking some input tokens, and exhibited some improvements in both conditioned and unconditioned generation tasks. At their turn, Lin et al. (2017) have proposed using relative rather than absolute rewards to ensure that the reward signals of different sentences remain distinguishable. Our work follows a similar idea to Lin et al. (2017), but aims to leverage a reward signal that correlates with, yet it is not identical to, a meaningful evaluation metric.

3 Model

As mentioned in Section 1, at a parity of training data, the quality of the generated sentences directly relates to a model's capacity (i.e., the order of magnitude of its size). Therefore, for a fair evaluation we adopt a widely cited benchmark called Texygen that has been used as reference by many other works (Nie et al., 2019; Caccia et al., 2020; Wang et al., 2020). Please note that this benchmark has streamlined all the implementations, but retained all the original algorithms faithfully. For the same reasons, we reuse

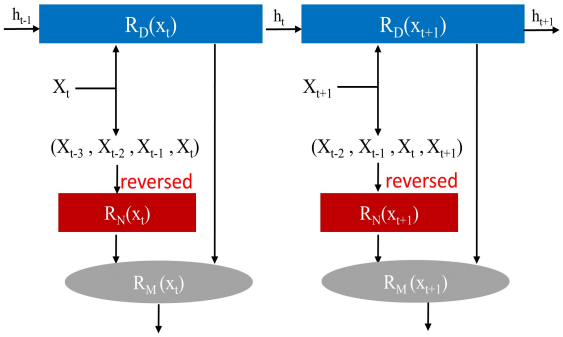

Figure 1: A sketch of the proposed model. The red boxes represent the computation of the n-gram matching (R_N). The blue boxes represent the reward from the unfolded recurrent discriminator (R_D). Eventually, the gray boxes represent the mixed reward (R_M).

most of its hyperparameters settings.

3.1 Generator

The generator G_θ that we have used for our work is a standard LSTM architecture (Hochreiter and Schmidhuber, 1997). However, our approach is architecture-agnostic, and larger models such as the transformer (Vaswani et al., 2017) can be equally employed. Typically, the generator is first pre-trained with the standard negative log-likelihood (NLL) loss to reduce the perplexity. After that, adversarial training is performed on the pre-trained model to fine-tune the generator over a chosen reward. The pre-training stage is usually beneficial with GAN-based models since the exploration space of adversarial training is much broader than that of NLL training (Chen et al., 2020). Indeed, if directly trained with the adversarial objective, models may fail to generate realistic sentences altogether.

3.2 Discriminator

We employ a discriminator, D_ϕ, with the same architecture as the generator except for the final linear layer, whose output dimension is 2 (real or generated) rather than the size of the vocabulary. Fedus et al. (2018) has shown that an LSTM-based discriminator can more effective than those based on CNNs because of its ability to assess the sequential nature of the sentence. The loss function of the discriminator can be expressed as:

$$\mathcal{L} = -\frac{1}{N_r}\sum_{r=1}^{N_r}\log D_\phi(X^r) - \frac{1}{N_g}\sum_{g=1}^{N_g}\log(1 - D_\phi(X^g)) \quad (1)$$

where X^r is a sentence from the training set ("real") and X^g is a generated sentence.

3.3 Reward

Our proposed training objective leverages two reward signals. The first, R_N, uses n-gram matching as reward. The second, R_D, is the signal from the discriminator. In addition, we have synthesized a third type of reward, R_M, by combining R_N and R_D. Figure 1 shows a sketch of our model, with emphasis on the reward signals. Thanks to the recurrent architecture of our discriminator, we are able to obtain a reward signal at every time step. Therefore, the reward signal from the discriminator for the k-th word generated by the generator can be simply expressed as:

$$R_D(x_k) = D_\phi(x_k) \quad (2)$$

where $D_\phi(x_k)$ measures the ability of word x_k to "fool" the discriminator.

For R_N, our basic assumption is that words with longer n-gram overlap with the training corpus should receive a higher reward than those with a shorter one. We use the notation $X_{k-n+1:k}$ to refer to the subsentence of length n ending at word x_k. Due to the sequential nature of the text, we should increase the order of n-grams from the left-end side to ensure the target token x_k is always included in the subsequences, like the reverse operation in the seq2seq model (Sutskever et al., 2014). Here, we set the maximum n to 4 as suggested by Papineni (2002). The n-gram matching, noted as $M(X_{k-n+1:k})$, is performed by comparing the target subsequence with the entire training corpus. If a match occurs, this process will output 1 and exit early, otherwise, it will return 0. Additionally, this process can be further accelerated by storing the already matched pairs in a look-up memory. Since the generation error increases with the length of the generated sequence (Bengio et al., 2015), for the first few tokens, that lack high-order n-grams, we can simply use a special $\langle pad \rangle$ token to fill the subsequence. Given that the $\langle pad \rangle$ token will never appear in the training corpus, this eventually ends up penalizing the initial tokens

which are easily learned and generated by the generator anyway, and force the generator to focus on later tokens.

To compute the total matching reward for word x_k, we use an equally-weighted average as suggested by Lavie and Agarwal (2007):

$$R_N(x_k) = \frac{1}{N} \sum_{n=1}^{N} M(X_{k-n+1:k}) \tag{3}$$

Note that, in alternative, it is possible to use different weights for the different orders of n-grams, so as to modulate the matching reward to different overlap preferences.

To form our third reward signal, R_M, from the combination of R_D and R_N, we have chosen to use a multiplication instead of linear interpolation, as in:

$$R_M(x_k) = R_N(x_k)\, R_D(x_k) \tag{4}$$

In this way, the final reward can be interpreted as a weighted value, where the value is the signal from the discriminator, and the weight depends on the largest length, n, of subsequence matching with the reference corpus. In case on no match, the reward will suitably reduce to 0. Overall, this scheme will encourage the generator to focus on generating sentences with longer n-grams overlapping. It should be noted that this is particularly important in unconditioned generation tasks, since there are no inputs or specific reference sentences to drive the training; rather, the main aim for this task is to generate grammatically- and lexically-correct sentences, and therefore more and longer overlapping n-grams will directly relate to the quality of the generated sentences, as we show in Section 4.

4 Experiments and results

4.1 Datasets and Experimental Set-Up

We have performed our experiments over three popular datasets for unconditioned text generation, namely COCO Image Captions, EMNLP2017 WMT News, and Chinese Poetry. The first two datasets consist of sentences of variable length, while the last dataset is made of sentences (verses) all of the same length. Further details are provided in the relevant subsections. To generate sentences, we can either always start from a fixed point or from a sampled point from a distribution, such as a variational autoencoder (Bowman et al., 2016). Here, we adopt the approach of Texygen, always starting from a special start-of-sentence token, $\langle sos \rangle$. We stop generating whenever another special token, $\langle eos \rangle$, is predicted, or a maximum pre-defined length has been reached. For generated sentences shorter than the maximum set length, we post-pad the sentence using a special pad token, $\langle pad \rangle$. During gradient backpropagation, we mask out all the $\langle pad \rangle$ tokens since they usually detract from the final performance. All models have been trained on the given training set, and evaluated on the corresponding test set. The number of generated sentences for the evaluation process has been set to the same size as the test set.

4.2 Results

As noted by Lavie and Agarwal (2007), recall-based metrics are meaningless if a task has no inputs and targets. Therefore, for evaluation we utilize the corpus BLEU score as a proxy for sentence quality, and the Self-BLEU score as a measure of the sentence diversity (more properly, sentence self-similarity; *i.e.*, the lower, the better) (Zhu et al., 2018). Following Texygen and virtually every other paper in this field, we report the results using the entire test set as the reference. The BLEU score is measured by comparing each generated sentence with each of the sentences in the given test set. This score reflects the ability of a model to generalize beyond the training set, by measuring the similarity between the generated sentences and those in the unseen test set. The Self-BLEU score is measured by comparing each generated sentence with each of the other generated sentences. While they are de-facto standards in the literature on unconditioned language generation, these two metrics are very sensitive to "best matches", and as such the results need to be discussed cautiously and integrated with qualitative analysis. To compare the quality and diversity trade-off of different methods, we follow Wang et al. (2020) and also report the *BLEU-5 F1* score. This score can be expressed as:

$$\text{BLEU-F1} = \frac{2 * \text{BLEU} * (1 - \text{Self-BLEU})}{\text{BLEU} + (1 - \text{Self-BLEU})} \tag{5}$$

In the following tables, the results referred to as *NLL* have been obtained by reproducing the experimental settings of Caccia et al. (2020) that carried

Model	BLEU ↑				Self-BLEU ↓				BLEU-F1 ↑
	B-2	B-3	B-4	B-5	B-2	B-3	B-4	B-5	B-5
SeqGAN*	0.745	0.498	0.294	0.180	0.950	0.840	0.670	0.489	0.266
MaliGAN*	0.673	0.432	0.257	0.159	0.918	0.781	0.606	0.437	0.248
RankGAN*	0.743	0.467	0.264	0.156	0.959	0.882	0.762	0.618	0.222
LeakGAN*	0.744	0.517	0.327	0.205	0.934	0.818	0.663	0.510	0.289
MaskGAN*	0.539	0.328	0.209	0.143	**0.752**	**0.516**	**0.378**	**0.293**	0.238
TextGAN*	0.593	0.463	0.277	0.207	0.942	0.931	0.804	0.746	0.228
NLL§	0.765	0.537	0.354	0.229	0.903	0.739	0.560	0.426	0.322
R_D	0.800	0.604	0.411	0.273	0.949	0.875	0.749	0.596	0.323
R_N	0.786	0.577	0.381	0.251	0.940	0.838	0.691	0.517	0.330
R_M	**0.829**	**0.653**	**0.463**	**0.314**	0.949	0.876	0.775	0.633	**0.338**

* results from Lu (2018).
§ reproduced with the best temperature ($\alpha = 1.25^{-1}$) provided by Caccia et al. (2020)

Table 1: The quality and diversity of generated sentences with the Image COCO dataset. Quality is represented by BLEU score, and diversity is represented by Self-BLEU score. We also include the BLEU-F1 score to compare the quality-diversity trade-off. NB: for the BLEU and BLEU-F1 scores, higher is better. For the Self-BLEU score, lower is better.

the cat are in a green long bowl .
a tv sits next to the ocean .
a large window in a room with a vehicle and reading a book .
a woman is sitting in a field surrounded by cars .
a airport filled wall sits on the back of a car
two people are preparing food in a kitchen .
a vintage quadruple propellor airplane looking

Table 2: A few randomly-selected sentences generated by the R_M model for the COCO Image Captions dataset, selected at the training iteration where the highest BLEU-5 score was obtained (potentially, the most challenging for diversity).

out a well-designed temperature sweeping. Therefore, we have used the best temperature parameter provided by their paper ($\alpha = 1.25^{-1}$) to reimplement the results under the same framework. For a fuller comparison, we also report the results on the same datasets from a pool of strong competitors including SeqGAN, MaliGAN (Che et al., 2017), RankGAN, LeakGAN, MaskGAN and TextGAN (Zhang et al., 2017).

4.2.1 COCO Image Captions

We first evaluate the performance of the proposed rewards on the COCO Image Captions dataset[1] (Lin et al., 2014). This dataset covers a vocabulary of 4.6K unique words, and the maximum sentence length is 37. The training and test sets contain $10,000$ sentences each. The results from all the compared models on this dataset are reported in Table 1. The scores show that our mixed reward model, R_M, has achieved the best performance in all the BLEU scores, which flags a better sentence quality compared to the other models. In terms of single rewards, R_D has achieved a higher BLEU score than R_N in all cases, yet always less than R_M. This gives evidence that combining n-gram matching information with a learned reward from a discriminator leads to synergy. Table 1 also shows that higher BLEU scores invariably come at a corresponding cost in diversity, as measured by the Self-BLEU score, as also previously reported in Caccia et al. (2020). This inescapable trade-off makes the comparison between different models difficult. For example, although MaskGAN has achieved the best (lowest) Self-BLEU scores, it has also reported the worst BLEU scores. Conversely, although our model has attained the highest BLEU scores, it has also suffered from a significant drop in Self-BLEU. Therefore, in order to compare the models based on a quality-diversity trade-off, following Wang et al. (2020), we report the BLEU-5 F1 score (longest n-gram BLEU) in the rightmost column of Table 1. The results show that on this dataset our

[1] http://cocodataset.org

Model	BLEU ↑				Self-BLEU ↓				BLEU-F1 ↑
	B-2	B-3	B-4	B-5	B-2	B-3	B-4	B-5	B-5
SeqGAN*	0.724	0.416	0.178	0.086	0.907	0.704	0.463	0.265	0.154
MaliGAN*	0.755	0.436	0.168	0.077	0.909	0.718	0.470	0.252	0.140
RankGAN*	0.686	0.387	0.178	0.086	0.897	0.677	0.448	0.298	0.153
LeakGAN*	0.835	0.648	0.437	**0.271**	0.938	0.821	0.668	0.510	**0.349**
MaskGAN*	0.265	0.165	0.094	0.057	**0.448**	**0.244**	**0.14**	**0.091**	0.107
TextGAN*	0.205	0.173	0.153	0.133	0.999	0.975	0.967	0.962	0.059
NLL§	0.850	0.573	0.323	0.182	0.848	0.588	0.327	0.172	0.298
R_D	0.839	0.553	0.303	0.160	0.881	0.623	0.361	0.214	0.267
R_N	0.836	0.602	0.371	0.215	0.873	0.703	0.482	0.490	0.302
R_M	**0.911**	**0.702**	**0.439**	0.237	0.941	0.829	0.668	0.421	0.336

* results from Lu (2018)
§ reproduced with the best temperature ($\alpha = 1.25^{-1}$) provided by Caccia et al. (2020)

Table 3: The quality and diversity of generated sentences with the EMNLP2017 WMT News dataset. Quality is represented by the BLEU score, and diversity is represented by Self-BLEU score. We also include the BLEU-F1 score to compare the quality-diversity trade-off. NB: for the BLEU and BLEU-F1 score, higher is better. For the Self-BLEU score, lower is better.

R_M model has been able to achieve the best quality-diversity trade-off. To show that our model in fact achieves appropriate diversity, Table 2 also provides a few randomly-selected sentences generated by the R_M model from the training iteration that scored the highest BLEU-5 score (potentially, the most challenging for diversity). The generated sentences show no evidence of major lack of diversity or "mode collapse". As such, we believe that the R_M model can be deemed as the best trade-off between quality and diversity for this dataset.

4.3 EMNLP2017 WMT News

We also evaluate our model on longer sentences with the EMNLP2017 WMT News dataset[2]. This dataset contains 5.3K unique words, and its maximum sentence length is 51. The training set and test sets consist of $260,000$ and $10,000$ sentences, respectively. The results from all the compared approaches on this dataset are shown in Table 3. Our mixed reward model, R_M, has outperformed all other models in the majority of BLEU scores (BLEU-2, BLEU-3, and BLEU-4), with the exception of BLEU-5 where it has been slightly outperformed by LeakGAN. In terms of single rewards, on this dataset R_N has achieved a higher BLEU score than R_D in most cases, which shows that these two reward signals have different performance over short and long text generation. For the quality-diversity trade-off, Table 3 shows the BLEU-5 F1 score of all the compared models. Interestingly, LeakGAN has reported the best performance in this metric and our R_M model has only ranked second. This is probably because LeakGAN was explicitly designed for long text generation, whereas our model is a general-purpose model that has no bias toward sentences of a specific length. Finally, for a qualitative analysis, Table 4 shows five random sentences generated by the R_M model at the iteration with the highest BLEU-5 score, showing, again, good diversity and also good overall quality. For comparison, Table 4 also shows five sentences from the NLL model (at the same random indexes, for an unbiased comparison): it can be seen that the sentences generated by our R_M model are significantly longer, and also more articulate in terms of sentence structure. This example epitomizes the behavior that the proposed reward is able to impart on the generator.

4.4 Chinese Poetry

Lastly, we evaluate our model on the Chinese Poetry dataset[3] of Zhang and Lapata (2014). For this experiment, we have chosen the section of the dataset containing five-word quatrain poems. This subset

[2] http://www.statmt.org/wmt17/

[3] https://homepages.inf.ed.ac.uk/mlap/Data/EMNLP14/

R_M:

975: she was declared not only used a fourth place in the whole , i knew in the car , " said the coach of those who had too living with a statement on a condition , according to the violent crime - point of the woman .
2785: how do you fight like a watch - fire for a very young woman , and i felt like that this time wasn ' t what they wouldn ' t have a sort of different feelings or the things do not actually have never been before .
5469: it will be four days more debates in the better of the strength of the top - elected .
9575: we have only made an opportunity to have to hear what ' s been also , which they know the ronald clinton ' s camp did not want to be able to continue to be tied with the context of the media and the kind of the proposals and we truly
9649: donald trump has warned that there will be possible thing , that the eu is doing all the women ' s campaign .

NLL:

975: the women , which is taken to the 10 queensland citizens protection extended to the execution pilot and couldn ' t net employment .
2785: in the list and on their page as a president , he speaks of the world back in past of what he will step up .
5469: she is a now coach , but the global crisis is no benefit resources to protect the father , very at federal authorities .
9575: a " rise in that - which warned if it said he didn ' t test growth under the presidency .
9649: when you ' re here when your children , i wouldn ' t keep home over the screen in your doctor .

Table 4: Five randomly-selected sentences generated by the R_M model (top) and by the *NLL* model (bottom) for the EMNLP2017 WMT News dataset.

consists of 4 lines in each poem, and 5 words in each line, resulting in a fixed poem length of 20 words in total. Since the most meaningful n-grams in Chinese poems are unigrams and bi-grams, in this experiment we follow the setting of Yu et al. (2017) and only use $n = 2$ for evaluating the BLEU, Self-BLEU and BLEU F1 scores. Also, for this dataset the maximum n-gram matching in Eq. 3 has correspondingly been set to 2. Table 5 shows that, in this case, R_N has achieved the highest BLEU score by a large margin, possibly because of the regular nature of the poems. However, R_N has also suffered from a correspondingly reduced diversity, indicated by the highest Self-BLEU score. Conversely, the R_D and NLL models have achieved better Self-BLEU scores, yet with much lower BLEU scores. The best trade-off between quality and diversity as expressed by BLEU F1 has been achieved by R_D. Overall, the results on this dataset differ from those on COCO Image Captions and EMNLP2017 WMT News, most likely because of its specific structural and semantic requirements (Zhang and Lapata, 2014).

Model	BLEU-2	Self-BLEU-2	BLEU-2 F1
NLL[§]	0.346	**0.377**	0.445
R_D	0.418	0.496	**0.496**
R_N	**0.635**	0.839	0.257
R_M	0.549	0.675	0.408

[§] reproduced with the best temperature ($\alpha = 1.25^{-1}$) provided by Caccia et al. (2020)

Table 5: BLEU, Self-BLEU, and BLEU F1 scores for the Chinese Poetry dataset.

5 Conclusion

In this paper, we have proposed a new training approach for generative adversarial networks for the task of unconditioned text generation. The approach leverages the policy gradient theorem to maximize a mixed reward signal (R_M), obtained from the multiplication of a reward based on n-gram matching with sentences in the training set (R_N) and a reward provided by a real-vs-generated discriminator (R_D). Our experimental results show that the proposed model has been able to achieve a comparable or better performance than its NLL baseline and several other GAN-based models in terms of sentence quality and quality-diversity trade-off. Since our n-gram matching scheme has margin for further optimization (such as, for instance, the use of different weights for the different n-gram orders, or the adoption of a collaborative matching scheme such as in METEOR (Lavie and Agarwal, 2007)), in the future we plan to explore other variants of the proposed approach, and also extend the evaluation to transformer-based language generators.

6 Acknowledgment

The first author is funded by the China Scholarship Council (CSC) from the Ministry of Education of P. R. China.

References

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam M. Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *NIPS*.

Antoine Bordes and J. Weston. 2017. Learning end-to-end goal-oriented dialog. *arXiv*, abs/1605.07683.

Samuel R. Bowman, L. Vilnis, Oriol Vinyals, Andrew M. Dai, R. Józefowicz, and S. Bengio. 2016. Generating sentences from a continuous space. In *CoNLL*.

Massimo Caccia, Lucas Caccia, William Fedus, Hugo Larochelle, Joelle Pineau, and Laurent Charlin. 2020. Language GANs falling short. *arXiv*, abs/1811.02549.

Noe Casas, José A. R. Fonollosa, and M. Costa-jussà. 2018. A differentiable bleu loss. analysis and first results. In *ICLR*.

Tong Che, Yanran Li, Ruixiang Zhang, R. Devon Hjelm, Wenjie Li, Y. Song, and Yoshua Bengio. 2017. Maximum-likelihood augmented discrete generative adversarial networks. *ArXiv*, abs/1702.07983.

Liqun Chen, K. Bai, Chenyang Tao, Yizhe Zhang, Guoyin Wang, Wenlin Wang, R. Henao, and L. Carin. 2020. Sequence generation with optimal-transport-enhanced reinforcement learning. In *AAAI*.

William Fedus, Ian J. Goodfellow, and Andrew M. Dai. 2018. MaskGAN: Better text generation via filling in the. *arXiv*, abs/1801.07736.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative adversarial networks. *arXiv*, abs/1406.2661.

Anirudh Goyal, Alex Lamb, Y. Zhang, Saizheng Zhang, Aaron C. Courville, and Yoshua Bengio. 2016. Professor forcing: A new algorithm for training recurrent networks. In *NIPS*.

Jiaxian Guo, S. Lu, Han Cai, W. Zhang, Y. Yu, and J. Wang. 2018. Long text generation via adversarial training with leaked information. In *AAAI*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9:1735–1780.

Matt J. Kusner and José Miguel Hernández-Lobato. 2016. GANs for sequences of discrete elements with the gumbel-softmax distribution. *arXiv*, abs/1611.04051.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *WMT@ACL*.

Tsung-Yi Lin, M. Maire, Serge J. Belongie, James Hays, P. Perona, D. Ramanan, Piotr Dollár, and C. L. Zitnick. 2014. Microsoft COCO: Common objects in context. In *ECCV*.

Kevin Lin, Dianqi Li, Xiaodong He, Ming-Ting Sun, and Zhengyou Zhang. 2017. Adversarial ranking for language generation. In *NIPS*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *ACL 2004*.

Sidi Lu, Yaoming Zhu, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Neural text generation: Past, present and beyond. *arXiv*, abs/1803.07133.

Weili Nie, Nina Narodytska, and Ankit B. Patel. 2019. RelGAN: Relational generative adversarial networks for text generation. In *ICLR*.

Kishore Papineni, S. Roukos, T. Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Alec Radford, Jeff Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Marc'Aurelio Ranzato, S. Chopra, Michael Auli, and W. Zaremba. 2016. Sequence level training with recurrent neural networks. *CoRR*, abs/1511.06732.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *ACL*.

Stanislau Semeniuta, Aliaksei Severyn, and Sylvain Gelly. 2018. On accurate evaluation of GANs for language generation. *arXiv*, abs/1806.04936.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv*, abs/1706.03762.

Guoyin Wang, C. Li, J. Li, Hao Fu, Yuh-Chen Lin, Liqun Chen, Yizhe Zhang, Chenyang Tao, Ruiyi Zhang, W. Wang, Dinghan Shen, Qian Yang, and L. Carin. 2020. Improving text generation with student-forcing optimal transport. *arXiv*, abs/2010.05994.

Ronald J. Williams. 2004. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256.

Y. Wu, M. Schuster, Z. Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, M. Krikun, Yuan Cao, Q. Gao, Klaus Macherey, J. Klingner, Apurva Shah, M. Johnson, X. Liu, Lukasz Kaiser, Stephan Gouws, Y. Kato, Taku Kudo, H. Kazawa, K. Stevens, George Kurian, Nishant Patil, W. Wang, C. Young, J. Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, G. Corrado, Macduff Hughes, and J. Dean. 2016.

Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv*, abs/1609.08144.

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

Xingxing Zhang and Mirella Lapata. 2014. Chinese poetry generation with recurrent neural networks. In *EMNLP*.

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

Yaoming Zhu, S. Lu, L. Zheng, Jiaxian Guo, W. Zhang, J. Wang, and Y. Yu. 2018. Texygen: A benchmarking platform for text generation models. *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*.