# Generative Adversarial Networks를 이용한 키워드 기반 문장 생성

박동주 ᴼ, 안창욱*
광주과학기술원 전기전자컴퓨터공학부
{toriving, cwan}@gist.ac.kr

# Sentence Generation from Keyword using Generative Adversarial Networks

Dongju Park<sup>O</sup>, Chang Wook Ahn*
School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology

**Abstract**

Generating coherent and meaningful text is important and applicable to a variety of fields. Not so long ago, Long Short-Term Memory (LSTM) Networks achieved effective performances in the area of Natural Language Generation (NLG). However, such method has trouble with *bias exposure*. Recent studies show that Generative Adversarial Networks (GAN) using reinforcement learning technique showed better performance than the currently existing LSTM model. Based on this, we propose a GAN model for generating sentences from a keyword and comparing it with the existing method. Experimental results on COCO dataset demonstrate that our proposed approach is more effective than the compared model.

## 1. Introduction

Natural Language Generation (NLG) has been one of the most difficult problems in Natural Language Processing (NLP) to date. Nevertheless, generating coherent and meaningful sentences plays an important part in many NLP applications such as dialogue systems, text summarization, and machine translation.

Recent studies show that the Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) [1] units can accomplish excellent performances for NLG work. A common approach is to train an RNN to maximize the log-likelihood of each real token in the training sequence given the prior observed tokens. However, such approach suffers from so-called *exposure bias* because of discrepancy between training and inference in the model. During the inference, the model sequentially generates the next token based on prior generated tokens; by contrast, when training, a ground-truth token is given. Consequently, the accumulation of errors caused by this difference increases with the length of the sequence [2].

Generative Adversarial Networks (GAN) [3] is emerging as one of the solutions to deal with the issue above. GAN typically consists of two neural network models that compete with each other. One model, called *Discriminator*, aims to distinguish real and generated (fake) data. The other is *Generator* that generates new data similar to the real one in order to deceive the discriminator. This approach using GAN has been successful in dealing with continuous data such as computer vision or image generation, but unfortunately, it has been difficult to deal with discrete data such as NLP. However, in recent research, discrete data can be handled by using SeqGAN [4] with reinforcement learning techniques.

In this paper, we propose a model of generating texts from keywords using GAN. We constructed a model based on SeqGAN and compared it with LSTM under the same conditions.

## 2. GAN for Sentence Generation from Keyword

Our model consists of generative and discriminative sub-models like conventional GAN. In order to deal with discrete data, we used a generative model of SeqGAN using LSTM, policy gradient, and Monte Carlo search. However, unlike SeqGAN, our model consists of two inputs, a sentence in the training set and an arbitrary noun in the sentence. An arbitrarily chosen noun is represented by a vector, which is used not only for the initial hidden state when learning a sentence containing the selected noun but also within the LSTM. Thus, the nouns represented by vectors sent into the LSTM are used with trainable variables during each gate and cell state update process. Once the learning is complete, you can generate a sentence by putting a noun that one thinks is a keyword for the generative model.

In the case of the discriminative model, we used Convolutional Neural Networks (CNN) [5] which has a great effect on text classification [6]. Like a generative model, it feeds sentences and noun as input. During training, there are two types of training data set: One is of real sentences and nouns that are extracted together from given sentences, and the other is of synthesized sentences and nouns that are input when generating the synthesized sentence. The noun is concatenated to the front of the sentence and used in the learning process. Also, these two kinds of training dataset are labeled to distinguish real and synthesized data.

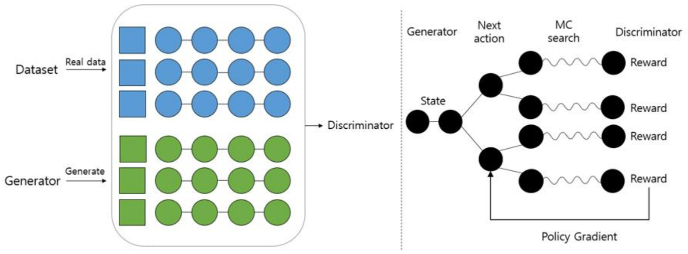


**Figure 1: The illustration of our proposed model. Left: D is trained with nouns and sentences of real and synthetic data by the generator. Right: The generator is trained by the final reward that is provided by the discriminator. Note that the rectangles mean a noun and the connected circles mean a sentence.**

Before adversarial training, our approach needs to be pre-trained as supervised learning. In the pre-training stage, the generator trains with a real dataset (sentence-noun pairs) to generate realistic data. The discriminator, on the other hand, trains to distinguish between real and fake, with a real dataset and a fake dataset (synthetic sentence generated by the generator with random noun-random noun pairs). After the pre-training stage, the generator and discriminator are adversarially trained. Adversarial training process is the same as SeqGAN, but the dataset is provided in the same way as the pre-training described above.

## 3. Experimental Results

A real dataset we choose is the COCO Image Captions Dataset [7] for the experiment, which consists of image-description pairs. We randomly select 20000 sentences for the training set and 1000 for the test set. The dataset is selected that its sentences only consist of 15 to 20 words. In addition, for the generative model sentence generation process, we only extracted the nouns in the dataset and created the word dictionary.

The pre-training model without adversarial training (MLE) is compared with our model. The model without adversarial training had only 250 pre-training epochs, and our proposed model had pre-training of 150 epochs and adversarial training of 100 epochs. After about 150 epochs, the Negative Log-Likelihood (NLL), the value of the loss function of the former, converges. In the latter case, NLL is further decreased because of adversarial training after 100 epochs. The learning curves shown in Figure 2 illustrate the comparison between two models.

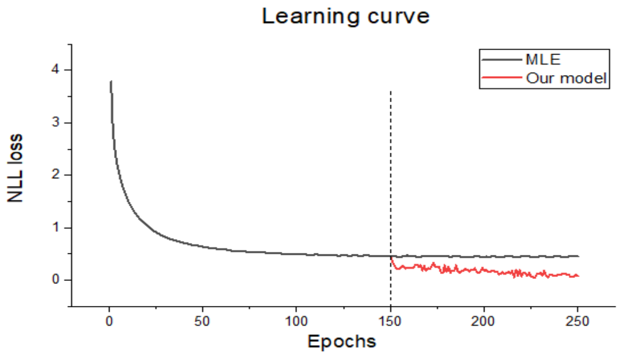


**Figure 2: Learning curve of two model. The vertical dash line stands for the end of pre-training of our model.**

**Table 1: The BLEU performance on COCO Dataset.**

| Method | BLEU-2 | BLEU-3 | BLEU-4 |
|---|---|---|---|
| MLE | 0.7050 | 0.5190 | 0.4775 |
| Our model | **0.7231** | **0.5307** | **0.4922** |

**Table 2: Example of real data and the generated sentence of two models. Note that models are trained on COCO Dataset**

| Sources | (Input noun) Example |
|---|---|
| Real data | - Three tiers of cupcakes, sodas, finger foods, and plastic wear on a table.<br>- A picture of a red bird that is standing on piece of wood next to peanuts.<br>- A little boy reading a book to an infant with his arm around the baby.<br>- A group of three giraffes eating greenery placed at the top of a wooden pole. |
| MLE | - (Sodas) Two sodas, four hot dogs, and two orders of cheese fries are on a table.<br>- (Picture) A very old picture of a steam train rounding a curve and coming down the track.<br>- (Arm) The brown a gray colored teddy bear sets with the number trunk in the ocean.<br>- (Giraffes) Road lights, electrical line, giraffes, sit, standing on brownish grass dock. |
| Our model | - (Sodas) Two sodas, four hot dogs, and an older foot painted at a laptop, one is carrying<br>- (Picture) This is a picture of a desk top with two computers on top of the TV.<br>- (Arm) The long haired fellow puts his arm around a picture under the bench.<br>- (Giraffes) Two giraffes and a zebra surrounded by green trees in an outdoor zoo. |

We also used BLEU as a way to evaluate and compare two models. The BLEU score [8] is originally designed to evaluate machine translation, although we employ it to evaluate the similarity of the generated sentence and the ground-truth sentence. We used BLEU-2, BLEU-3, and BLEU-4 to evaluate sentence generation performance. As a result, our model is better than the MLE model. In all evaluations, the proposed method is more effective than the existing method, and the results show that our model generates better sentences when the same keyword is fed into each model. The results are shown in Table 1 and 2.

## 4. Conclusion

In this paper, we propose a model for generating sentences from keyword using GAN based on SeqGAN. The model is fed not only sentences but also nouns during the generator and discriminator training, and then pre-training before performing adversarial training with each other. Experimental results of our approach showed the better performance than that of the generating sentences by providing nouns to the MLE model. Our model can be extended to generating sentences with multiple keywords rather than a single keyword and can be extended to producing sentences that have a similar meaning to a given keyword or contain words that are close in distance to the keyword. Also, if we use a more efficient model for sentence generation than SeqGAN, we obtain better results.

**Acknowledgements**

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2015R1D1A1A02062017).

## Reference

[1] Hochreiter, S., and J. Schmidhuber. Long short-term memory, *Neural Computation*, 9: 1735-80. 1997.

[2] Bengio, S., Vinyals, O., Jaitly, N., and Shazeer, N. Scheduled sampling for sequence prediction with recurrent neural networks. In *NIPS*, 1171-79. 2015.

[3] Goodfellow, I., Pouget-Abadie, J., Mehdi, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *NIPS*, 2672-80. 2014.

[4] Yu, L., Zhang, W., Wang, J., and Yu, Y. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. In *AAAI*, 2852-58. 2017.

[5] Krizhevsky, A., Sutskever, I., and Hinton, G. E. *Imagenet* classification with deep convolutional neural networks. In *NIPS*, 1097-105, 2012.

[6] Zhang, X., and LeCun, Y. Text understanding from scratch, *arXiv preprint arXiv:1502.01710*. 2015.

[7] Chen, X., Fang, H., Lin, T.-Y., Vedantam, R., Gupta, S., Dollár, P., and Zitnick, C. L. Microsoft COCO captions: Data collection and evaluation server, *arXiv preprint arXiv:1504.00325*. 2015.

[8] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. BLEU: a method for automatic evaluation of machine translation. In *ACL*. 311-18. 2002.