# Sentimental Style Transfer in Text with Multigenerative Variational Auto-Encoder

Mehedi Hasan Palash
*Department of*
*Computer Science and Engineering*
*Shahjalal University of*
*Science and Technology*
*evonloch@gmail.com*

Partha Protim Das
*Department of*
*Computer Science and Engineering*
*Shahjalal University of*
*Science and Technology*
*meparthaprotim@gmail.com*

Summit Haque
*Department of*
*Computer Science and Engineering*
*Shahjalal University of*
*Science and Technology*
*summit.haque@gmail.com*

2019 International Conference on Bangla Speech and Language Processing (ICBSLP) 978-1-7281-5241-7/20/$31.00 ©2020 IEEE 10.1109/ICBSLP47725.2019.201508

***Abstract*—Style transfer is an emerging trend in the fields of deep learning's applications, especially in images and audio data this is proven very useful and sometimes the results are astonishing. Gradually styles of textual data are also being changed in many novel works. This paper focuses on the transfer of the sentimental vibe of a sentence. Given a positive clause, the negative version of that clause or sentence is generated keeping the context same. The opposite is also done with negative sentences. Previously this was a very tough job because the go-to techniques for such tasks such as Recurrent Neural Networks(RNNs) [1] and Long Short-Term Memories(LSTMs) [2] can't perform well with it. But since newer technologies like Generative Adversarial Network(GAN) and Variational Auto-Encoder(VAE) are emerging, this work seem to become more and more possible and effective. In this paper, Multi-Genarative Variational Auto-Encoder is employed to transfer sentiment values. Inspite of working with a small dataset, this model proves to be promising.**

***Index Terms*—text, style, style-transfer, vae, sentiment-transfer**

## I. Introduction

In this fast-growing era of deep learning, understanding and transferring of stylistic attributes in numerous fields have been proven very fruitful. Especially in the case of images and other visual forms of data, neural networks have worked wonders in capturing the style elements and manipulating them [3]. This study explores the transfer of styles in linguistic expressions. The possibilities of text style transfer are limitless. It can be used in making customized chatbots that can interact with humans as any human would do. This could come in handy in different organizations and even in personal interests. Moreover, generating parallel data can be another implication of textual style transfer. This is a very important aspect because in many style transfer job the main problem becomes getting the parallel data. Transferring the style from one stream of text to another can solve this problem easily and reliably with much less effort.

Since dealing with textual data is very different from pictorial data, transferring styles is much of a challenge here. Mostly because in transferred images, the slight distortions can be overlooked, which is not the case for linguistic data. Here the model has truly to understand the context and do things according to that. This work is based on a basic stylistic attribute, that is-sentiment. We choose to work with converting a very specific sentiment in human interactions- the positivity and negativity of sentences. We take a sentence or clause with a positive vibe and then change it to a negative one. This is a primitive step towards general style transfer in text. But if this basic problem can be solved, we can hope to get closer to more human-like machine-generated text.

We used Multi Generator Variational Auto-Encoder(VAE) to achieve our goal. We also tried vanilla Auto-Encoder, but this does not work as good as VAEs because they are continuous. On the other hand, both mean and variance are taken into account in VAEs and that simple technique helps get a better result, for this expands the window of opportunity for getting more diverse outputs. Therefore we get more relevant outputs. We use two different decoders to get two types of response. One for positive and another for negative styled output.

## II. Related Work

Style transfer with non parallel text has been exercised extensively in recent years. The most influential style transfer work would be of Gatys et al. [3]. It is shown here that the style of images can be transferred. And this and several such works inspire to build some machine similar to them which would work for textual data. Zhang et al. [4] uses CNN for this task, which face many problems mostly because text and image are different in nature and should not be treated alike. Therefore, Neural Machine Translations (NMTs) come into play [5] [6]. Parallel data is used to let the model learn from them. But this is a bit problematic to find parallel texts, and sometimes it is even not possible to get in the post-NMT era of text style transfer, Xu et al. [6], Fu et al. [7], Li et al. [8], Shen et al. [9], Prabhumoye et al. [10] introduce newer techniques like GANs [11] and VAEs to perform this job, in an unsupervised way. That is-without the help of parallel data. There have been many different techniques to evaluate the success of these approaches. Shen et al. [9] uses sentiment modification, word substitution cipher decypherment and word order recovery as human verification factors. They infer styles from a sentences and its original style indicator. A style dependent decoder is used to render them. Moreover, a brilliant technique-cross generated sentences is used to gather more information. Hu et al. [12] employs Variational Auto-Encoders (VAEs) and attribute discriminators to generate sentences, whose attributes are controlled by trained disentangled latent representation of them. Here yelp and amazon datasets are used and evaluated by humans to get more accurate response.

## III. Dataset

### A. Data Source

We used a dataset on bangla sentences. 4600 comments are collected from prothom-alo [13] news. Every comment has 6 options for tagging. They are 'Surely Negative', 'Slightly Negative', 'Neutral', 'Slightly Positive', 'Surely Positive'. We get the data tagged three times by volunteers for making the tags more reliable. Some examples are shown in figure 1

| | | |
|---|---|---|
| 0 | লিখার সময় পারলে সত্য লিখার অভ্যাস শিখুন। | কিছুটা নেতিবাচক |
| 1 | এটা কেন হচ্ছে? সংশ্লিষ্ট সকলের ডিপ্রেশনের ফলে?... | কিছুটা নেতিবাচক |
| 2 | আমাদের দেশের স্বাভাবিক অর্থনৈতিক গতিপ্রবাহকে ব... | কিছুটা নেতিবাচক |
| 3 | চুরি নয় লুটপাট। | নিশ্চিত নেতিবাচক |
| 4 | ইসলামী ব্যাংকের বর্তমান অবস্থা দেখে মনে হয় শাস... | নিরপেক্ষ |

Fig. 1. Sample data

For the purpose of using the tags appropriately to learn them, we replaced the tags by some numerical values. These are assigned in accordance with their strength. The more the negative a sentence seem to be are assigned more negative values, and the same but opposite goes for the positive sentences. ['Surely Negative', 'Slightly Negative', 'Neutral', 'Slightly Positive', 'Surely Positive'] = [-2,-1,0,1,2]

We then sum up three tags for each instance of the dataset. If the sum is less than zero, we marked it as negative and if greater than zero mark it as positive comment. For example: Suppose a comment has three tags : 'Surely Negative', 'Neutral', 'Slightly Positive'. So the sum for this comment will be : -2 + 1 + 0 = -1. So ultimately we treat this as a negative comment. After eliminating duplicates and unusable sentences, we get 2500 negative comments and 2500 positive comments.

### B. Preprocessing

When we work with text data, we have to read stream of characters. Single characters normally don't mean anything. If we combine them together carefully, they will make sense. Tokenizing means split the stream of characters such that we can consider them as semantic unit of language. Tokenization [14] also removes the punctuation marks. Here is an example of tokenization shown in figure 2

```
Sentence:
"রাজপণ্ডিত হব মনে আশা করে । সপ্তশ্লোক ভেটিলাম রাজা গৌড়েশ্বরে ।।"

After tokenization:
['রাজপণ্ডিত', 'হব', 'মনে', 'আশা', 'করে', '।', 'সপ্তশ্লোক', 'ভেটিলাম', 'রাজা', 'গৌড়েশ্বরে', '।', '।']
```

Fig. 2. Tokenizing Sentences

We used Natural Language ToolKit(NLTK) for tokenization. We fixed the sequence length to 20. For sentences smaller than 20, we padded them with a fixed string.

## IV. Model Overview

We have used a variational auto-encoder (Doersch et al [15]) with two generators in our model. Normally every variational auto-encoder has two parts. An encoder and a decoder. First, the encoder part maps the input sequence to a latent representation z. Then, the generator part takes samples and makes them saturated with the desired style, which is positive and negative in our case.

We can consider the encoder as a neural network and it takes datapoint x and its output is a hidden representation z. Then, another neural network called generator, samples from z and produces the desired output. z preserves the context of the input and decoder is used for mixing the desired attribute's properties.

In this model depicted in figure 3, two generators are used-one for positive and another for negative. The positive generator is specialized to modify sentences with more positivity. The other is for generating negative sentences. First of all, the input sentence passes through an encoder network. From the encoder network, a parameter of distribution $Q(z|x)$ where $x$ is the vectorized input sentence. The latent vector z is got from the stated distribution. Now this $z$ is the information holder of $x$. The decoders use $z$ to recreate the sentence with a varied style.

While training, the positive decoder and the negative decoder are separately used so that they can learn about a specific style. That is-when dealing with positive data, we train the negative generator. Again we train the positive generator with the negative data to make it capable of generating positive sentences.

## V. Result

We have used human evaluation for determining our model's accuracy. As there is not enough software resource for evaluating style transfer in Bangla. Human evaluation has been used in extensively in the validation of recent linguistic tasks such as in the works of Shen et al [9].

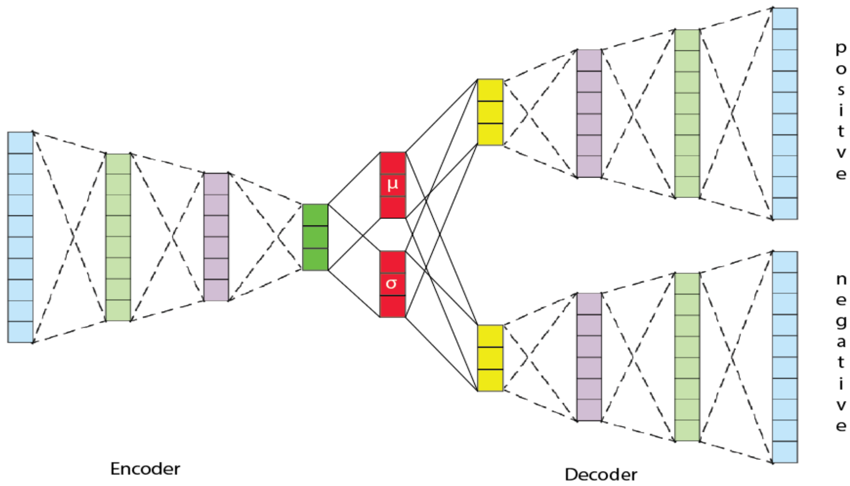


Fig. 3. High Level view of the model

We have measuerd the following parameters for determining the success of an output:

- ***Grammatical Correctness*** This metric is used to see how much the model can understand the structure of the language.
- ***Context Similarity*** Whether the output could capture the context is expressed as this.
- ***Polarity*** This is the most important point in our task-whether the output convey positive or negative vibe.

We have copied the input of the model and used it as the output to get a baseline result. Since human comments are used as input baseline output, the output has hundred percent grammatical correctness and context similarity with input. But it has not achieved the correct polarity because the negativity of a positive sentence is small, and vice versa.

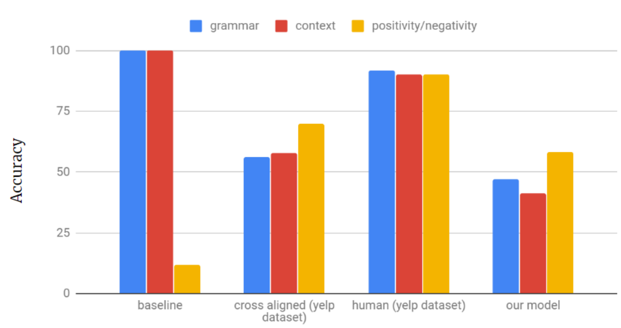


Fig. 4. Comparing our results with state of the art models

Comparing with the human tested results with the yelp dataset, all the parameters(grammar, context, positivity/negativity) are much higher than all the state of the art models. This means that machines couldn't be able to beat humans in this respect. But among the best performing models, cross aligned technique is proven to be the best. Since our dataset is eerily small and in bangla, we couldn't achieve such greateness. Nonetheless, we have got closer to that model in grammar and sentiment transferability, though we need to focus on the fact that context-wise we are far from expectation.

## VI. Analysis

The sentiment transferability and grammatical correctness of our model are not so far from the model prescribed by Shen et al.(2017) [9]. But the context preservation is not satisfactory. The dataset we have used has variety of contexts. Some are about politics, some are about sports and so on. But Shen et al.(2017) [9] used a standard dataset from amazon product review. So, the context preservation of our model is not as good as their model. On the other hand if we could use data from a single context then it might have performed better.

A sample output generated by our model is shown here in figure 5. Here, we can see that for the given positive sentence,

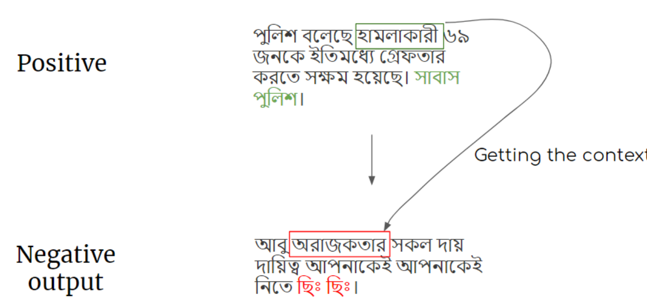


Fig. 5. Positive to generated Negative output

our model tried to get the context(green boxed) and then staying within the context, it tries to generate totally opposite a sentence. Similarly, if a negative sentence is fed into the machine as in figure , the context is still captured and then the style of this negativity is changed, and it becomes a positive sentence.

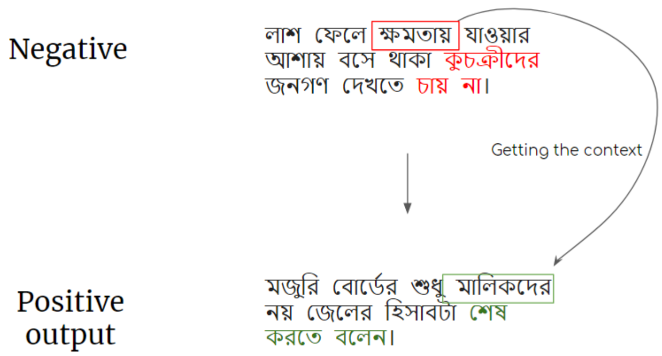


Fig. 6. Negative to generated Positive output

## VII. Conclusion

### A. Discussion

Transferring style in text is not much explored field. It's still a newer area than other explorable fields like images and audios. Yet it is so much promising. Because there are so many things we can do with textual data. With the power to generate new sentences is alone a huge task to do with a machine. And if we can apply our customized styles and personalization to it, this sounds even more wonderful, because up until now this was solely a humane capability.
We explore this vast field's only a small portion-translating negative sentences into positive ones and translating positive sentences into negative ones. Our model is able to succeed in some cases, the accuracy we get is 53.2%. This might not be very good result, but our model couldn't do much well because of shortage of data. If we could get hold of much larger dataset, this model could achieve more, mostly because our model is very much data dependent.

### B. Future Work

Working with sentimental values is an initial step in textual style transfer process. In this paper only the positivity and negativity of a sentence are exploited. There are many other aspects of styles in a sentence that can be explored in this manner, like-gender, tense, political stand etc. If these basic style-bearing aspects can properly be understood and make the machine understand, it will have many interesting applications, like-personalized chatbots and even making an universal language translator.

## VIII. Acknowledgement

We would like to thank the NLP group of Shahjalal University of Science and Technology for the necessary insight and expertise that greatly assisted the research.

## References

[1] A. Karpathy, "The Unreasonable Effectiveness of Recurrent Neural Networks," May 2015. [Online]. Available: http://karpathy.github.io/2015/05/21/rnn-effectiveness/
[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[3] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2414–2423.
[4] X. Zhang and Y. LeCun, "Text understanding from scratch," *arXiv preprint arXiv:1502.01710*, 2015.
[5] J. M. Hughes, N. J. Foti, D. C. Krakauer, and D. N. Rockmore, "Quantitative patterns of stylistic influence in the evolution of literature," *Proceedings of the National Academy of Sciences*, vol. 109, no. 20, pp. 7682–7686, 2012.
[6] W. Xu, A. Ritter, B. Dolan, R. Grishman, and C. Cherry, "Paraphrasing for style," in *Proceedings of COLING 2012*, 2012, pp. 2899–2914.
[7] Z. Fu, X. Tan, N. Peng, D. Zhao, and R. Yan, "Style transfer in text: Exploration and evaluation," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
[8] J. Li, R. Jia, H. He, and P. Liang, "Delete, retrieve, generate: A simple approach to sentiment and style transfer," *arXiv preprint arXiv:1804.06437*, 2018.
[9] T. Shen, T. Lei, R. Barzilay, and T. Jaakkola, "Style transfer from non-parallel text by cross-alignment," in *Advances in neural information processing systems*, 2017, pp. 6830–6841.
[10] S. Prabhumoye, Y. Tsvetkov, R. Salakhutdinov, and A. W. Black, "Style transfer through back-translation," *arXiv preprint arXiv:1804.09000*, 2018.
[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
[12] Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing, "Toward controlled generation of text," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 1587–1596.
[13] "Prothom Alo," https://www.prothomalo.com/, accessed: 2019-07-31.
[14] "Tokenization," https://nlp.stanford.edu/IR-book/html/htmledition/tokenization-1.html , accessed: 2019-07-10.
[15] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.