IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 2, 2019, accepted July 16, 2019, date of publication July 25, 2019, date of current version August 8, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2931036

Emotional Text Generation Based on Cross-Domain Sentiment Transfer

RUI ZHANG, ZHENYU WANG, KAI YIN, AND ZHENHUA HUANG
Department of Software Engineering, South China University of Technology, Guangzhou 510006, China

Corresponding author: Zhenyu Wang (wangzy@scut.edu.cn)

This work was supported in part by the Science and Technology Program of Guangzhou, China, under Grant 201802010025, and in part by the University Innovation and Entrepreneurship Education Fund Project of Guangzhou under Grant 2019PT103.

ABSTRACT Emotional intelligence plays an important role in human intelligence and is a recent research hotspot. With the rapid development of deep learning techniques in recent years, several neural network-based emotional text generation methods have been investigated. However, the existing emotional text generation approaches often suffer from the problem of requiring large-scale annotated data. Generative adversarial network (GAN) has shown promising results in natural language generation and data enhancement. In order to solve the above problem, this paper proposes a GAN-based cross-domain text sentiment transfer model, which uses annotated data from other domains to assist in the training of emotional text generation network. By combining adversarial reinforcement learning with supervised learning, our model is able to extract patterns of sentiment transformation and apply them in emotional text generation. The experimental results have shown that our approach outperforms the state-of-the-art methods and is able to generate high-quality emotional text while maintaining the consistency of domain information and content semantics.

INDEX TERMS Emotional text generation, adversarial learning, sentiment transfer.

I. INTRODUCTION

Emotional intelligence is one of the important parts of human intelligence, which has received extensive attention and research interests in the field of natural language process. Accurately recognizing and understanding emotions in text facilitates tasks such as human-computer interaction, opinion analysis, and community discovery. Nowadays neural network based approaches have made great progress in text sentiment analysis tasks [1]. However, the task of emotional text generation is still very difficult compared to the success of sentiment classification [2], [3].

Traditional emotional text generation methods are mainly based on hand-crafted rules and utterance templates, and often fail to handle with complicated cases. With the rapid development of deep learning, it has become possible to generate diverse texts using neural language model. In recent years, several studies have attempted to endow text generation model with emotional intelligence, such as generating conversational responses with specific emotions [4]–[6], or synthesizing positive/negative product reviews [7].

The associate editor coordinating the review of this manuscript and approving it for publication was Biju Issac.

Text style transformation is a method of generating emotional text that has attracted widespread attention [8]–[14]. The basic idea is to edit the style attribute of a sentence while keeping the semantic content, as shown in Figure 1. Due to the lack of a clear definition of the style in natural language, the existing research works usually equate the style with the sentiment. However, these text sentiment transformation approaches typically require large-scale corpus with emotional labels for neural model training, which makes it difficult to apply these methods to domains that lack of sufficient annotated data.

Generative Adversarial Network (GAN) [15] is a novel data enhancement solution that uses a discriminative network to guide the training process of the generator, which has been widely applied in text generation task. In GANs, the generator is trained to generate text that can fool the discriminator, which prevents the discriminator from determining whether the text is sampled from the real corpus or generated by the generator. Since the generator and discriminator are trained alternately and independently, the generative network can produce high quality text with a similar distribution to the corpus after a sufficient number of training iterations.

Inspired by the research of unsupervised cross-domain image generation [16], this paper propose the GAN-based

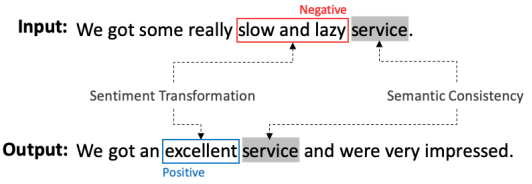

FIGURE 1. An example of text style transformation.

cross-domain text sentiment transfer model to solve the problem of insufficient annotated data in emotional text generation in some certain domains. The proposed model consists of an emotional text generator, a sentiment discriminator and a domain discriminator. The generator that based on encoder-decoder architecture is trained to extract semantic information from the original input and to output a sentence with original content and specific emotion. The discriminators are introduced to force the generator to learn emotional patterns from corpus in other domains (refer to as auxiliary domains) and apply these patterns to current domain. By combining supervised learning with adversarial reinforcement learning, our model achieves great performance in the emotional text generation task.

We conduct experiments on the Yelp dataset and the Amazon dataset. The experimental results show that our approach is competitive with the state-of-the-art methods in the single-domain sentiment transfer task, and is superior to the existing models in the cross-domain sentiment transfer task.

The major contributions of this paper are summarized as follows:

- We present a novel idea of learning emotional patterns from cross-domain data to solve the problem of insufficient annotated data in emotional text generation tasks.
- We propose a new approach based on adversarial reinforcement learning for emotional text generation. The experiments performed on two datasets demonstrate the efficacy and superiority of our approach.

The rest of the paper is arranged as follows. In Section II we review the current research progress of emotional text generation. In Section III we formally describe the task of cross-domain text sentiment transfer and present the details of our method. We outline the experiments and analysis in Section IV and V. We close with our conclusions and a discussion of future work in Section VI.

II. RELATED WORK

Natural language generation (NLG) is an important research topic in the field of natural language processing. With the widespread use of deep learning techniques in NLG, many researchers have attempted to generate emotionally colored text in language generation task. There are two main ideas in the existing research of emotional text generation: (1) neural machine translation (NMT) based methods, which use the large-scale annotated conversation corpus to train an end-to-end dialogue model, and (2) style transfer based methods, which edit a sentence with a certain style (sentiment) into another sentence with a specific style.

The NMT-based emotional text generation methods, which "translate" the conversation input into a response with specific emotion, are typically applied in dialogue generation tasks. Several NMT-based methods have been proposed recently. Zhou *et al.* [4] introduced emotional chatting machine (ECM), which generate emotional responses by adopting emotion category embedding, internal emotional memory and external memory. Ghosh *et al.* [5] proposed Affect-LM which generate emotionally colored conversational text in five specific affect categories with varying affect strengths. Zhang *et al.* [6] regarded generating conversational responses with different emotion types as a multitasking problem. Sun *et al.* [17] used the SeqGAN model to improve the ability of conversation model to generate emotional responses.

The emotional text generation methods based on text style transfer can be regarded as a sentence editing task, which modifies the style of the sentence and keeps other semantic content unchanged [8]. Due to the lack of a clear definition of the style in natural language, the existing research work usually equates the style with the sentiment. Generally, text style transfer models are trained using non-parallel corpora in specific domain. Shen *et al.* [9] investigated a method to achieve style transfer by using the cross-alignment of the latent representation in hidden layers. Fu *et al.* [10] regarded style transfer as a multi-task learning problem, and investigated multi-decoder and style-embedding methods for style transfer. Xu *et al.* [11] introduced a cycled reinforcement learning approach, which consists of two modules: the neutralization module to remove emotional words and extract non-emotional semantic information, and the emotionalization module to add sentiment to semantic content. John *et al.* [12] proposed a disentangled representation learning approach, using auxiliary loss and adversarial learning to enforce the separation of style and content latent spaces. By combining the potential representations of style and content information, the model is capable of generating text with the corresponding style.

Different from the aforementioned encoding-decoding based methods, Guu *et al.* [13] investigated a prototype-then-edit approach which first samples a prototype sentence from the training corpus, then edit this sentence using neural network to change its style. Li *et al.* [14] proposed a retrieve-and-delete based method for text style transfer task, which deletes the words associated with the original attribute from the sentence, and retrieves new phrases associated with the target style, then synthesizes new text using a neural model.

The existing emotional text generation approaches usually requires a large-scale corpus with emotional annotation for model training. However, some domains do not have such available corpus. In order to combat the problem of insufficient data in specific domain, this paper proposes a cross-domain sentiment transfer model based on generative adversarial learning that use the annotation data from other domains to assist in model training.

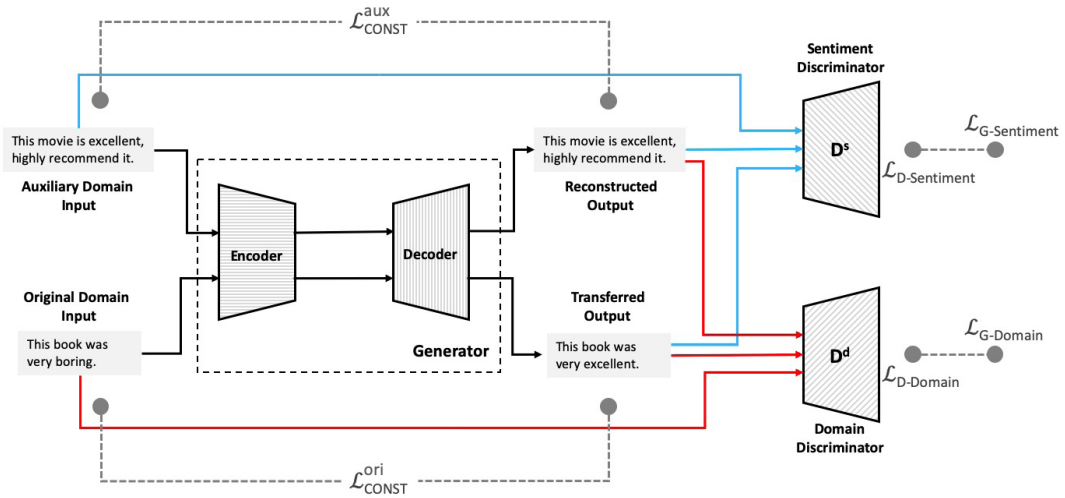

FIGURE 2. Overview of our approach. The input/output are drawn with solid lines, losses with dashed lines. The generator is alternately trained by supervised learning (using both auxiliary and original domain text), and adversarial reinforcement learning (using original domain text).

III. PROPOSED APPROACH

The motivation for proposing the cross-domain text sentiment transfer method is to solve the problem that we lack of well-annotated training data for emotional text generation. Existing methods generally assume that large-scale corpora with emotional annotations are available for model training. However, for some domains that suffering from insufficient annotated data problem, it is difficult to train emotional text generation models using these deep learning methods.

Inspired by the idea of unsupervised cross-domain image generation [16], we propose a text sentiment transfer model, which can be trained using annotated corpus from other domains (refer to as auxiliary domains) and fine-tuned on the original domain, to combat the problem of lack of available labeled training data for emotional text generation. In other words, we expect the model to learn emotional patterns from auxiliary domain data and apply these patterns to emotional text generation in current domain. We formally describe the cross-domain text sentiment transfer task as follows:

Given a text set $\mathbf{C}^{\text{ori}} = \{X_1^{\text{ori}}, X_2^{\text{ori}}, \cdots\}$ in the origin domain with arbitrary emotion types, and a text set $\mathbf{C}^{\text{aux}} = \{X_1^{\text{aux}}, X_2^{\text{aux}}, \cdots\}$ in the auxiliary domain with sentiment $\hat{s}$, the task is to learn a mapping function $G : X \rightarrow Y$ that takes sentence X as input and generates a sentence Y with emotion type $\hat{s}$, where Y and X have the similar content information. Since the model is trained using non-parallel corpus from two different domains, the model is expected to retain the domain-related information in the original sentence X when synthesizing the sentence Y.

The overall framework of our cross-domain sentiment transfer model is illustrated in Figure 2. The model includes a generator based on the encoder-decoder framework and two discriminators based on neural classification networks. The generator takes a sentence X as input and converts X to a sentence Y with the target emotion $\hat{s}$. The discriminator is used to distinguish between "fake" sentences generated by the generator and "real" sentences sampled from the training corpus. The sentence Y synthesized by the generator should be as close as possible to the distribution of real natural language, and can fool the style discriminator and the domain discriminator.

This section begins with an overview of the proposed approach. Details of the sequence-to-sequence based generator and the cross-domain sentiment transfer method are presented in Section III-A and Section III-B. We further discuss the details of generative adversarial learning for sequence generation in Section III-C.

A. SEQ2SEQ BASED GENERATOR

The generator is a sequence-to-sequence (Seq2Seq) based neural network, which consists of an encoder and a decoder. For each input sentence $X = \{x_1, x_2, \cdots, x_m\}$, the encoder $\mathbf{E}$ encodes X into a sequence of hidden representations $\mathbf{h} = \{h_1, h_2, \cdots, h_m\}$. Then, the decoder $\mathbf{D}$ takes h_m as the initial state and decode a sequence $Y = \{y_1, y_2, \cdots, y_n\}$ by predicting the next word with the highest probability of generation.

We employ the gated recurrent unit (GRU) [18] network as the encoder and the decoder. A GRU unit consists of the following components:

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \tag{1}$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \tag{2}$$

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t]) \tag{3}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \tag{4}$$

where $\tilde{h}_t$ denotes the intermediate state which computes the candidate activation; h_t represents the activation of GRU at time t; r_t is a reset gate, which controls the effect of the previous activation h_{t-1} on the current candidate activation state $\tilde{h}_t$; and z_t is the update gate that controls the update process

of the current activation based on the previous activation h_{t-1} and the candidate activation $\tilde{h}_t$.

B. CROSS-DOMAIN SENTIMENT TRANSFER NETWORK

To learn the emotional text generation model using cross-domain, non-parallel corpus, we propose the cross-domain text sentiment transfer model based on generative adversarial learning.

First, we use the aforementioned Seq2Seq-based generator G as the generative network. Second, two RNN-based classifiers are introduced as discriminators to distinguish between sentences generated by the generator G and sentences sampled from training corpora. The generator is expected to synthesize text that can fool the discriminators. Finally, we train the generator and discriminators alternately in each iteration in a supervised manner and tune the parameters of the generator using adversarial reinforcement learning. After training on large-scale corpus, the generator is able to perform sentiment transformation on the original sentence. The basic structure of our model is as follows:

Given an original input X, we expect the generator to synthesizes a sentence Y which reserves the content of X and also contains the target sentiment information. In other words, we wish to learn a function $G : X \rightarrow Y$ which minimizes the combined loss:

$$\mathcal{L} = \mathcal{L}_{\text{CONST}} + \mathcal{L}_{\text{GAN}} \tag{5}$$

where $\mathcal{L}_{\text{CONST}}$ indicates the loss of content consistency, and $\mathcal{L}_{\text{GAN}}$ is the adversarial loss.

We have adopted two different learning strategies to make use of training data from different domains:

For the auxiliary domain text, the generator is trained in a supervised manner using sequence-aggregated cross-entropy loss, and only the content consistency loss $\mathcal{L}_{\text{CONST}}$ is considered, denoted as:

$$\mathcal{L}^{\text{aux}}_{\text{CONST}} = -\sum_{t=1}^{n} \log p_\theta(y_t|h_m, y_1, \cdots, y_{t-1}) \tag{6}$$

where θ denotes the parameters of the generator, and $p_\theta(y_t|\cdot)$ indicates the probability of generating the target token y_t.

For the original domain text, both $\mathcal{L}_{\text{CONST}}$ and $\mathcal{L}_{\text{GAN}}$ are taken into consideration. We tune the parameters of the generator using adversarial learning and reinforcement learning. In this case, we adopt a modified Bilingual Evaluation Understudy (BLEU) algorithm [19] to evaluate the content consistency loss $\mathcal{L}_{\text{CONST}}$ of the generated sentence. Our modifications are simple yet effective: directly mask the ''sentiment indicator'' tokens in the generated sentence Y (considered as candidate) and the input sentence X (considered as reference), then calculate a standard BLEU score. Thus, the consistency loss for original domain text is calculated as:

$$\mathcal{L}^{\text{ori}}_{\text{CONST}} = 1 - \text{BLEU}(Y') \tag{7}$$

where Y' is the masked generated sentence. It is worth noting that we still need additional knowledge to determine if a token is the ''sentiment indicator''. In practice, we use sentiment lexicons provided by Wilson *et al.* [20] to identify these words.

In order to make the generator capable of editing the original input into a sentence with specific sentiment, we introduce two discriminative components for adversarial learning: the sentiment discriminator and the domain discriminator. The training procedure of adversarial learning can be considered as a minimax game between the generator and the discriminators. Each discriminative model is trained to distinguish between the ''fake'' text synthesized by the generative model and the ''real'' text sampled from training corpus. At the same time, the generative model is trained to generate text that can fool the discriminative models. Therefore, $\mathcal{L}_{\text{GAN}}$ can be regarded as a trade-off between the discriminative loss $\mathcal{L}_{\text{D}}$ and the generative loss $\mathcal{L}_{\text{G}}$.

The sentiment discriminator is used to evaluate the sentiment transfer performance of the generator. It is trained by minimizing the classification loss $\mathcal{L}_{\text{D-Sentiment}}$, which is calculated as:

$$\begin{aligned}\mathcal{L}_{\text{D-Sentiment}} = &-\mathbb{E}_{Y\sim\mathbf{C}^{\text{aux}}}[\log D^s_\phi(Y)] \\ &-\mathbb{E}_{Y\sim\mathbf{G}(X^{\text{aux}})}[\log(1 - D^s_\phi(Y))] \\ &-\mathbb{E}_{Y\sim\mathbf{G}(X^{\text{ori}})}[\log(1 - D^s_\phi(Y))]\end{aligned} \tag{8}$$

where D^s is the sentiment discriminator and ϕ represents its parameters. $\mathbf{C}^{\text{aux}}$ denotes the auxiliary domain corpus; $\mathbf{G}(X^{\text{aux}})$ denotes the set of sentences generated by generator G with text in auxiliary domain as input, while $\mathbf{G}(X^{\text{ori}})$ denotes the ones with text in original domain as input.

Meanwhile, generator G is trained to fool the discriminator D^s. In other words, generator G attempts to generate sentences with the same sentiment as the text in the auxiliary domain corpus. Formally, the objective for generator G is to minimize the loss $\mathcal{L}_{\text{G-Sentiment}}$, denoted as:

$$\mathcal{L}_{\text{G-Sentiment}} = -\mathbb{E}_{Y\sim\mathbf{G}(X^{\text{ori}})}[\log D^s_\phi(Y)] \tag{9}$$

Since the above-mentioned adversarial learning process may cause the generator G to be more prone to generate tokens related to the auxiliary domain, it is necessary to introduce an additional component to avoid this situation. Similar to the sentiment adversarial network, we adopt a domain adversarial network to evaluate and enhance the domain information preservation capability of the generator. Formally, the domain discriminator is trained by minimizing the classification loss $\mathcal{L}_{\text{D-Domain}}$, which is:

$$\begin{aligned}\mathcal{L}_{\text{D-Domain}} = &-\mathbb{E}_{Y\sim\mathbf{C}^{\text{ori}}}[\log D^d_\eta(Y)] \\ &-\mathbb{E}_{Y\sim\mathbf{G}(X^{\text{ori}})}[\log(1 - D^d_\eta(Y))] \\ &-\mathbb{E}_{Y\sim\mathbf{G}(X^{\text{aux}})}[\log(1 - D^d_\eta(Y))]\end{aligned} \tag{10}$$

where D^d represents the domain discriminator and η denotes its parameters. Similarly, the generator G is optimized by minimizing the loss $\mathcal{L}_{\text{G-Domain}}$, which is calculated as:

$$\mathcal{L}_{\text{G-Domain}} = -\mathbb{E}_{Y\sim\mathbf{G}(X^{\text{ori}})}[\log D^d_\eta(Y)] \tag{11}$$

We consider the estimated probability of being "real" text which calculated by the discriminators as reward. Thus, minimizing $\mathcal{L}_{\text{G-Sentiment}}$ and $\mathcal{L}_{\text{G-Domain}}$ can be regarded as a process of maximizing rewards, and the optimization problem of generative network G can be seen as a reinforcement learning problem. However, adversarial learning for sequential data is still difficult. We will discuss these details in the next section.

C. SEQUENCE GENERATIVE ADVERSARIAL LEARNING

Unlike the generative adversarial learning in the image domain, there are two major challenges in using generative adversarial networks for sequence generation. On the one hand, the classical GANs are designed to generate continuous real-value data, and are not suitable for generating sequence of discrete tokens. This is due to the fact that in GANs, the loss from discriminator with respect to the outputs by generator will direct the generator to slightly change its output value to make the generated results more realistic. In the process of text generation, however, it is difficult to achieve because in the limited vocabulary space there is probably no corresponding words to reflect such change. On the other hand, the discriminator network typically only scores the entire generated sequence, which make it hard for the generator to explicitly learn which discrete token has a greater impact on the quality of the generated results.

Following the adversarial learning approach for dialogue generation proposed by Li *et al.* [21] and the SeqGAN model proposed by Yu *et al.* [22], we regard text generation as a sequential decision making process, and optimize this process by reinforcement learning. For each time step t, we consider the generated tokens $\{y_1, y_2, \cdots, y_{t-1}\}$ as state s, and the next token y_t as action a. We use the reward estimate approach introduced in SeqGAN, which calculate the value of intermediate action via Monte Carlo search with a roll-out policy. Therefore, the sentiment transfer reward $r_{\text{sentiment}}$ and the domain preservation reward r_{domain} are calculated as:

$$r_{\text{sentiment}} = \begin{cases} \frac{1}{K}\sum_{k=1}^{K} D_{\phi}^{s}(Y_{1:n}^{k}) & \text{for} \quad t < n. \\ D_{\phi}^{s}(Y_{1:t}) & \text{for} \quad t = n. \end{cases} \tag{12}$$

$$r_{\text{domain}} = \begin{cases} \frac{1}{K}\sum_{k=1}^{K} D_{\eta}^{d}(Y_{1:n}^{k}) & \text{for} \quad t < n. \\ D_{\eta}^{d}(Y_{1:t}) & \text{for} \quad t = n. \end{cases} \tag{13}$$

where $Y_{1:n}^{k} = \{y_1, \cdots, y_t, \cdots, y_n\} \in \text{MC}^{G_\beta}(Y_{1:t}; K)$ is sampled based on the roll-out policy G_β and the current state, n is the length of output sequence Y.

Then we optimize the parameters of the generative model using the REINFORCE algorithm. After training using large-scale corpus, the generative model is capable of generating sentences with specific emotion in a certain domain.

To put it all together, the training process of the presented model is illustrated in Algorithm 1.

Algorithm 1 GAN-Based Cross-Domain Sentiment Transfer

Require: Training steps γ_1 and γ_2; Generator G; Sentiment Discriminator D^s; Domain Discriminator D^d; Unlabeled text dataset $\mathbf{C}^{\text{ori}} = \{X_1^{\text{ori}}, X_2^{\text{ori}}, \cdots, X_m^{\text{ori}}\}$from the original domain; Annotated text dataset $\mathbf{C}^{\text{aux}} = \{X_1^{\text{aux}}, X_2^{\text{aux}}, \cdots, X_n^{\text{aux}}\}$ from the auxiliary domain with target sentiment $\hat{s}$;

Initialize G, D^s and D^d with random weights;
Pre-train G using MLE on $\mathbf{C}^{\text{ori}}$ and $\mathbf{C}^{\text{aux}}$;
Generate fake texts F^{ori} and F^{aux} on the two domains using generator G;
Pre-train D^s and D^d using $\{\mathbf{C}^{\text{ori}}, \mathbf{C}^{\text{aux}}, F^{\text{ori}}, F^{\text{aux}}\}$;
repeat
  for g-steps **do**
    Sample batch X^{aux} from the auxiliary domain dataset and generate fake texts F^{aux} using $G(X^{\text{aux}})$;
    Update G by minimizing Eq. (6);
    Sample batch X^{ori} from the original domain dataset and generate fake texts F^{ori} using $G(X^{\text{ori}})$;
    Update G by minimizing Eq. (7);
    for γ_1-steps **do**
      for t in 1:T **do**
        Compute $r_{\text{sentiment}}$ by Eq. (13);
      end for
      Update G via REINFORCE algorithm;
    end for
    for γ_2-steps **do**
      for t in 1:T **do**
        Compute r_{domain} by Eq. (13);
      end for
      Update G via REINFORCE algorithm;
    end for
  end for
  for d-steps **do**
    Generate fake texts F^{ori} and F^{aux} on the two domains using generator G;
    Update D^s using $\{\mathbf{C}^{\text{aux}}, F^{\text{aux}}, F^{\text{ori}}\}$ by minimizing Eq. (8);
    Update D^d using $\{\mathbf{C}^{\text{ori}}, F^{\text{ori}}, F^{\text{aux}}\}$ by minimizing Eq. (10);
  end for
until model converges

IV. EXPERIMENTAL SETUP

A. DATASET

We experiment with the Yelp review dataset under a single-domain setup, and conduct experiments on the Amazon review dataset to compare the performance of the models in cross-domain sentiment transfer task. The details of the two datasets are as follows:

Yelp Review Dataset. Since the baseline methods are text sentiment transformation models for single-domain,

following previous work [9] we use the Yelp restaurant review dataset[1] to evaluate the performance of our model in single domain text sentiment transfer task. The Yelp dataset contains 444,101 reviews for train, 63,483 for validation and 126,670 for test. The maximum review length is 15 words.

Amazon Review Dataset. Experiments are also conducted on the Amazon review dataset [23], and we compare these methods using cross-domain text. This dataset contains a large number of Amazon product reviews with user rating tags and is divided into 24 categories. We choose the ''Books'' and the ''Movie and TV'' reviews for our experiment. Similar to previous work [11], we preprocess the dataset with the following steps. First, we consider the reviews with rating below three as negative reviews, and reviews with rating above three as positive reviews. Since the user rating annotation is provided at the documentation level and our method focuses on sentence-level text generation, we use the Natural Language Toolkit[2] (NLTK) to split the review paragraph into sentences. Then, we use the corresponding rating to label these sentences, and filter out the ones that exceed 20 words. Finally, we train a sentiment classifier and filter out reviews with classification confidence below 0.75, as the sentiment-level weak annotation mentioned above result in some noise data. After preprocessing, about 10.6M reviews for ''Books'' category and 2.4M reviews for ''Movie and TV'' category are retained. We use the negative reviews in the ''Books'' category as the original domain text, and the positive reviews in the ''Movie and TV'' category as the auxiliary domain text. In other words, given a negative review for a book, we expect the final model to generate a positive review for this book.

B. MODEL DETAILS

We implement both the encoder and the decoder as 2-layer GRUs, with word embedding dimension of 100 and hidden layer dimension of 128. The maximum utterance length is set to 20. All parameters are initialized by sampling from a uniform distribution over -0.1 and 0.1. In order to train the generator more effectively, we first pre-train it as an autoencoder using Maximum Likelihood Estimate(MLE), then tune the model parameters using our approach. The generator is pre-trained for 5 epochs on both the original and the auxiliary datasets, and the word embedding vectors are learned during pre-training.

The discriminators for adversarial reinforcement learning are implemented as single layer GRU with hidden size of 128. The training step γ_1 and γ_2 are set to 3 and 5 respectively. We use the Adam optimizer [24] to train our model.

C. BASELINES

We compare our approach with the state-of-the-art systems below:

Cross-Alignment Auto-Encoder (CAAE): This work is proposed by Shen *et al.* [9], which investigated a method to achieve style transfer by using the cross-alignment of the latent representation in hidden layers.

Unpaired Sentiment Translation (UST): This method is proposed by Xu *et al.* [11]. They introduced a cycled reinforcement learning approach for sentiment transfer task. Their model consists of two modules: the neutralization module which removes emotional tokens and extracts non-emotional content information, and emotionalization module which adds sentiment to the semantic content.

D. METRICS

We evaluate the generation results of our approach from the following aspects: (1) sentiment transfer strength, (2) content preservation, and (3) quality of generated text. We apply several automated methods to evaluate the first two aspects. Since automatically evaluating the quality of generated text is still a difficult task, we report the manual analysis results to further confirm the performance of the models in Section V-C. The automatic evaluation metrics are as follows:

Sentiment Transfer Strength: Transfer strength is the most common and basic metrics [25]. We adopt a separate LSTM-sigmoid network to predict the sentiment category of the generated text, following the previous works by Fu *et al.* [10] and John *et al.* [12]. The sentiment transfer strength could be defined as:

$$\text{Transfer Strength} = \frac{N_{\text{True}_s}}{N_{\text{Total}}} \tag{14}$$

where N_{True_s} denotes the number of test case which is successfully transferred to target sentiment, and N_{Total} represents the size of the test set. The sentiment classifier achieves the accuracy of 93% on the Yelp dataset and the accuracy of 90% on the Amazon dataset.

Domain Preservation Strength: Since training with cross-domain data may cause the decoder to generate words that is unrelated to the original domain, we introduce another classifier to evaluate the model's capabilities of preserving domain information. Similarly, we use another LSTM classifier to determine whether the generated sentence belongs to the original domain or the auxiliary domain, denoted as:

$$\text{Preservation Strength} = \frac{N_{\text{True}_d}}{N_{\text{Total}}} \tag{15}$$

where N_{True_d} is the number of generated sentences classified as original domain text. The domain classifier achieves the accuracy of 84% on the Amazon dataset.

Cosine Similarity: The similarity of sentence vectors is a simple yet effective measure of sentence similarity, as introduced in [10]. We calculate the sentence embedding by concatenating the *min*, *max* and *mean* of its word embeddings. Then we compute the cosine similarity between the input sentence and the generated sentences, defined as:

$$\text{Sim}_{\text{cosine}}(s_1, s_2) = \frac{\overrightarrow{v_1} \cdot \overrightarrow{v_2}}{\|\overrightarrow{v_1}\| \, \|\overrightarrow{v_2}\|} \tag{16}$$

[1] https://www.yelp.com/dataset
[2] http://www.nltk.org

TABLE 1. Automatic evaluation results.

Experiment Set	Model	Sentiment Transfer Strength	Domain Preservation Strength	Cosine Similarity	Word Overlap
Yelp	CAAE	76.4	-	**98.8**	**87.3**
	UST	**83.4**	-	96.5	84.8
	Ours (w/o D^d)	81.7	-	95.0	83.9
Amazon	CAAE	40.7	65.4	**92.6**	74.7
	UST	51.8	57.2	86.7	61.8
	Ours	**69.2**	**71.3**	92.2	**75.4**

where v_1 and v_2 are the sentence embeddings corresponding to the input sentence s_1 and the generated sentence s_2.

Word Overlap: Word overlap is another effective metric for content preservation [12]. We further refine this assessment approach by excluding sentiment-specific words. In particular, we exclude the sentiment words from the lexicon introduced by Wilson *et al.* [20] for our evaluation. The word overlap similarity is defined as:

$$\text{Sim}_{\text{overlap}}(s_1, s_2) = \frac{\text{COUNT}(k_{s_1} \cap k_{s_2})}{\text{COUNT}(k_{s_1} \cup k_{s_2})} \tag{17}$$

where k_{s_1} and k_{s_2} are the unigram words corresponding to sentences s_1 and s_2.

V. RESULTS AND ANALYSIS

A. AUTOMATIC EVALUATION ANALYSIS

To verify the generation capability of our proposed model, we conduct two sets of experiments on different datasets: the Yelp dataset for single-domain text sentiment transformation and the Amazon dataset for cross-domain text sentiment transformation. For each set of experiments, we run each model 5 times and evaluate the performance of the models for each metrics, and calculate the final results for each model through averaging. The experimental results are shown in Table 1.

We first perform the single-domain text sentiment transformation experiment on the Yelp dataset and compare it with the two baseline models. We did not use the domain discriminator D^d when conducting experiment on the Yelp dataset since this dataset does not contain cross-domain data. Experimental results on the Yelp dataset show that our model outperforms the CAAE model in terms of sentiment transfer strength and is competitive with the UST model. Since the CAAE model produces more failed results (most of which are the same as the input), it scores higher on cosine similarity and word overlap metrics.

We further conduct cross-domain text sentiment transformation experiment on the Amazon dataset to verify the domain information preservation capabilities of these models. As can be seen in Table 1, our presented model is able to distinguish words from different domain, and effectively preserve domain information while performing sentiment transformation. In the sentiment transfer strength and the domain preservation strength metrics, our model is significantly better than the baseline models, and also has an increase of 0.7 over the best baseline model on the word overlap metrics.

The reason why the baseline models perform poorly on the Amazon (cross-domain) dataset is that these models lack of external domain knowledge. When training with both original and auxiliary domain data, it is difficult for these models to learn whether a word is related to a particular domain or emotion. In particular, the UST model contains a classification module for identifying emotional words. Due to the different data distribution between the original domain and the auxiliary domain corpus, the classifier incorrectly confuses the domain-specific words and sentiment words, which further leads to the wrongly editing in the text generation process. For example, for the input sentence "What a boring book", the UST model trained using cross-domain data may synthesize an output "What a boring movie". In the above example, the UST model erroneously edits the domain-specific word "book" while retaining the emotional word "boring".

Although the model presented in this paper does not explicitly use external knowledge to inform the model which word is related to the original domain, the generator can still learn the relationship between words and domains via adversarial learning. Therefore, the model can better distinguish between domain-specific words and emotional words, thus obtaining better scores in sentiment transfer strength and domain preservation strength metrics.

B. PERFORMANCE CORRELATION ANALYSIS

In adversarial learning, the setting of training times γ has a crucial impact on generated results. We further evaluated the performance of the model under different setting of training times γ to select the optimal model hyperparameter.

We first determine the setting range of γ_i by running pre-experiment, and then combine the two sets of parameters. We evaluate the performance of our model with different combinations of parameter setting, where $\gamma_1 \in \{3, 5, 7\}$ and $\gamma_2 \in \{3, 5, 7\}$, as illustrated in Figure 3.

Figure 3(a) shows that the score of sentiment transfer strength is basically negatively correlated with the score of domain preservation strength under different settings of γ_i. Therefore, we should find the appropriate parameter settings so that the model can perform well on both metrics. Figure 3(b) illustrates that the model exhibits a negative correlation between the score of sentiment transfer strength and the score of word overlap, which is to say, while improving the ability of the model to modify the emotion of the text, the capability of the model to retain the semantics information of the original sentence will be reduced. Figure 3(c) indicates that the score of domain preservation strength is positively correlated with the performance of the word overlap metrics. This suggests that improving the model's ability to preserve

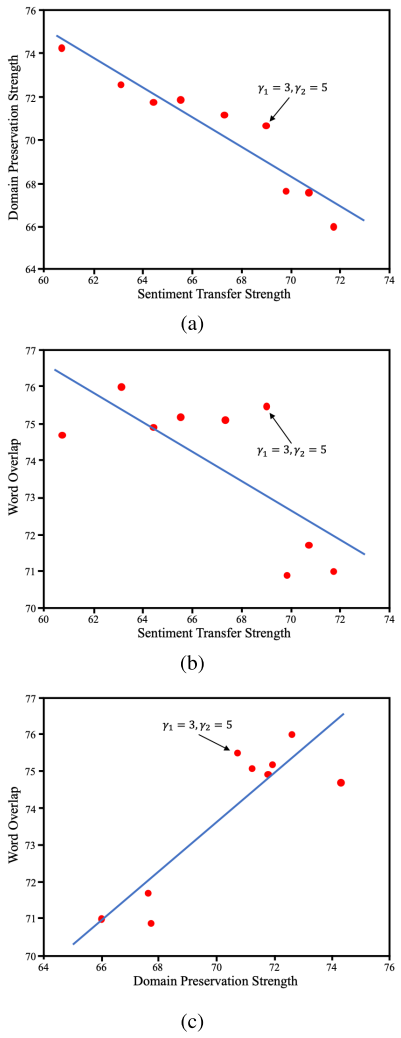

FIGURE 3. Score correlation of sentiment transfer strength, domain preservation strength, and word overlap under different training times settings.

domain information helps the model generate sentences that are more capable of maintaining the original semantics.

Finally, we choose $\gamma_1 = 3, \gamma_2 = 5$ as the final setting for our model, since the model achieves more acceptable results on each evaluation metrics under this setting.

C. HUMAN EVALUATION ANALYSIS

In order to better evaluate the proposed approach, we also conduct manual evaluation to assess the quality of generated sentences. We recruit three annotators for human evaluation experiments. Each annotator is asked to rank the generated results according to the following criteria: (1) emotional accuracy, (2) domain accuracy, and (3) linguistic fluency. The results of manual ranking can be considered as relative scores. We use the following method to calculate the score: 3 points for the first place, 2 points for the second and 1 point for the last. Since different models may produce the same results for some test case, we allow the rankings to be juxtaposed.

We extracted 150 sentences from the test set and manually evaluated the sentiment transformation results for each model. The results of human evaluation are shown in Table 2. We calculate the Fleiss' kappa as the statistical measure of inter-rater consistency, the average score is 0.716.

TABLE 2. Human evaluation results.

Experiment Set	Model	Avg Score
Yelp	CAAE	1.93
	UST	**2.07**
	Ours	1.99
Amazon	CAAE	2.05
	UST	1.71
	Ours	**2.24**

TABLE 3. Typical successes and failures in the experimental results.

	Examples
(1)	**Input:** This is a disappointing book. **Output:** This is a amazing book.
(2)	**Input:** The story lines are cliche and uninteresting. **Output:** The story are amazing and :)
(3)	**Input:** I didn't even finish it , it was sooo boring! **Output:** I didn't even finish it , it was sooo refreshing! **(Failed)**
(4)	**Input:** I won't read any more books in this series. **Output:** I won't see any more actor in this series. **(Failed)**

We also randomly sample some cases from the generated results of our model and perform manual analysis. Typical successes and failures in the experimental results are presented in Table 3.

As can be seen, the presented model can well identify the explicit emotional words in the sentence, and replace these words with the target emotional expressions. However, it is still difficult for the model to capture the metaphorical emotional expressions, which further leads to the failure of the emotional text generation task. For example, in case (3), the model has identified the sentiment word "boring", but failed to recognize the metaphorical expression "didn't even finish it", thus resulting in emotionally contradictory output.

Another typical failure is because the model confuses domain-specific words and sentiment words, which in turn leads to the generation of out-of-domain information, as shown in case (4). The results of automatic evaluation and manual analysis indicate that our model performs better on this issue because we apply additional modules to force the model to distinguish domain information.

Although we did not introduce additional network structures to learn domain knowledge, the presented model still achieve acceptable results in cross-domain text sentiment transfer task. We believe that there are two main reasons: on the one hand, the word vectors learned during neural network training can reflect the role of specific words in emotional expression. Words with similar meanings will be close to each other in the embedding space, which enables the association of similar emotional expression in different domains. On the other hand, there is a certain similarity between the "Book" domain and the "Movie and TV" domain. For example, there are some shared domain-specific words such as "storyline", and similar emotional expression such as "amazing". This enables the model to generate sentences for original domain without external domain knowledge.

VI. CONCLUSION AND FUTURE WORK

In this paper, we propose a GAN-based cross-domain text sentiment transfer approach, which is used to solve the problem of lacking of annotated data in emotional text generation tasks. By combining adversarial reinforcement learning and supervised learning, the proposed model can learn sentiment transformation patterns from auxiliary domain data and apply them to emotional text generation. The experimental results demonstrate the efficacy of our approach. In future work, we will consider more complex networks to capture semantic associations between different domains to enhance the quality of generated emotional texts. In addition, we will also apply this model to the data-to-text generation task.

REFERENCES

[1] T. Al-Moslmi, N. Omar, S. Abdullah, and M. Albared, "Approaches to cross-domain sentiment analysis: A systematic literature review," *IEEE Access*, vol. 5, pp. 16173–16192, 2017.

[2] L. Zhang, S. Wang, and B. Liu, "Deep learning for sentiment analysis: A survey," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 8, no. 4, p. e1253, 2018.

[3] M. Yang, W. Yin, Q. Qu, W. Tu, Y. Shen, and X. Chen, "Neural attentive network for cross-domain aspect-level sentiment classification," *IEEE Trans. Affect. Comput.*, to be published.

[4] H. Zhou, M. Huang, T. Zhang, X. Zhu, and B. Liu, "Emotional chatting machine: Emotional conversation generation with internal and external memory," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 730–738.

[5] S. Ghosh, M. Chollet, E. Laksana, L.-P. Morency, and S. Scherer, "Affect-LM: A neural language model for customizable affective text generation," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2017, pp. 634–642.

[6] R. Zhang, Z. Wang, and D. Mai, "Building emotional conversation systems using multi-task Seq2Seq learning," in *Proc. Nat. CCF Conf. Natural Lang. Process. Chin. Comput.* Cham, Switzerland: Springer, 2017, pp. 612–621.

[7] A. Radford, R. Jozefowicz, and I. Sutskever, "Learning to generate reviews and discovering sentiment," 2017, *arXiv:1704.01444*. [Online]. Available: https://arxiv.org/abs/1704.01444

[8] W. Choi, S. J. Choi, S. Park, and S.-J. Lee, "Adversarial style transfer for long sentences," in *Proc. Int. Conf. Electron., Inf., Commun. (ICEIC)*, Jan. 2019, pp. 1–3.

[9] T. Shen, T. Lei, R. Barzilay, and T. Jaakkola, "Style transfer from non-parallel text by cross-alignment," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6830–6841.

[10] Z. Fu, X. Tan, N. Peng, D. Zhao, and R. Yan, "Style transfer in text: Exploration and evaluation," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 663–670.

[11] J. Xu, X. Sun, Q. Zeng, X. Ren, X. Zhang, H. Wang, and W. Li, "Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach," 2018, *arXiv:1805.05181*. [Online]. Available: https://arxiv.org/abs/1805.05181

[12] V. John, L. Mou, H. Bahuleyan, and O. Vechtomova, "Disentangled representation learning for non-parallel text style transfer," 2018, *arXiv:1808.04339*. [Online]. Available: https://arxiv.org/abs/1808.04339

[13] K. Guu, T. B. Hashimoto, Y. Oren, and P. Liang, "Generating sentences by editing prototypes," *Trans. Assoc. Comput. Linguistics*, vol. 6, pp. 437–450, Jul. 2018.

[14] J. Li, R. Jia, H. He, and P. Liang, "Delete, retrieve, generate: A simple approach to sentiment and style transfer," in *Proc. Conf. North Amer. Assoc. Comput. Linguistics, Hum. Lang. Technol. (NAACL-HLT)*, vol. 1, New Orleans, LA, USA, Jun. 2018, pp. 1865–1874.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[16] Y. Taigman, A. Polyak, and L. Wolf, "Unsupervised cross-domain image generation," in *Proc. 5th Int. Conf. Learn. Represent. (ICLR)*, Toulon, France, Apr. 2017.

[17] X. Sun, X. Chen, Z. Pei, and F. Ren, "Emotional human machine conversation generation based on SeqGAN," in *Proc. 1st Asian Conf. Affect. Comput. Intell. Interact. (ACII Asia)*, May 2018, pp. 1–6.

[18] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder—Decoder for statistical machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1724–1734.

[19] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, 2002, pp. 311–318.

[20] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," in *Proc. Hum. Lang. Technol. Conf. Empirical Methods Natural Lang. Process.*, 2005, pp. 347–354.

[21] J. Li, W. Monroe, T. Shi, S. Jean, A. Ritter, and D. Jurafsky, "Adversarial learning for neural dialogue generation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 2157–2169.

[22] L. Yu, W. Zhang, J. Wang, and Y. Yu, "SeqGAN: Sequence generative adversarial nets with policy gradient," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 2852–2858.

[23] R. He and J. McAuley, "Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering," in *Proc. 25th Int. Conf. World Wide Web*, 2016, pp. 507–517.

[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980

[25] R. Mir, B. Felbo, N. Obradovich, and I. Rahwan, "Evaluating style transfer for text," 2019, *arXiv:1904.02295*. [Online]. Available: https://arxiv.org/abs/1904.02295

RUI ZHANG received the B.S. degree from the School of Software Engineering, South China University of Technology, Guangzhou, China, in 2015, where he is currently pursuing the Ph.D. degree. His research interests include natural language generation, text mining, and sentiment analysis.

ZHENYU WANG received the Ph.D. degree from the Department of Computer Science, Harbin Institute of Technology, in 1993. He is currently the Dean of the School of Software, South China University of Technology, and the Director of the Guangdong Provincial Social Media Processing and Engineering Center. His research interests include natural language processing, text mining, and social network analysis.

KAI YIN received the B.S. degree in computer science from Anhui University, Hefei, China, in 2018. Since 2018, he has been a Graduate Student with the South China University of Technology. His research interest includes dialog systems.

ZHENHUA HUANG was born in Anhui, China, in 1993. He received the B.S. degree from the School of Software Engineering, South China University of Technology, Guangzhou, in 2014, where he is currently pursuing the Ph.D. degree. He has been selected as a Visiting Scholar and a Young Big Data Scientist with the University of California at Irvine, Irvine, in the Program of IBM-CSC Y-100. He has authored more than five articles. His research interests include social computing, sentiment analysis, and deep learning.