Emotion Style Transfer with a Specified Intensity Using Deep Reinforcement Learning

Anonymous ACL submission

Abstract

Text style transfer is a widely explored task in natural language generation which aims to change the stylistic properties of the text while retaining its style-independent content. In this work, we propose the task of emotion style transfer with a specified intensity in an unsupervised setting. The aim is to rewrite a given sentence, in any emotion, to a target emotion while also controlling the intensity of the target emotion. Emotions are gradient in nature, some words/phrases represent higher emotional intensity, while others represent lower intensity. In this task, we want to control this gradient nature of the emotion in the output. Additionally, we explore the issues with the existing datasets and address them. A novel BART-based model is proposed that is trained for the task by direct rewards. Unlike existing work, we bootstrap the BART model by training it to generate paraphrases so that it can explore lexical and syntactic diversity required for the output. Extensive automatic and human evaluations show the efficacy of our model in solving the problem.

1 Introduction

Text style transfer is a popular task in natural language generation that controls a certain attribute or stylistic property in the output text, e.g. sentiments (Shen et al. (2017), Liu et al. (2021)), formality (Rao and Tetreault, 2018), politeness (Madaan et al., 2020), emotions (Helbig et al., 2020), linguistic style based of authors (Syed et al., 2020).

Early works in style transfer was supervised in nature with the availability of parallel data (Carlson et al. (2017), Jhamtani et al. (2017)). Creating a parallel corpus is time-consuming and expensive. Thus recent focus has been on unsupervised style transfer, working on a non-parallel corpus. Broadly, non-parallel style transfer can be divided into three categories: 1) Explicit style-content disentanglement, 2) Implicit style-content disentanglement, and 3) without disentanglement (Hu et al., 2020).

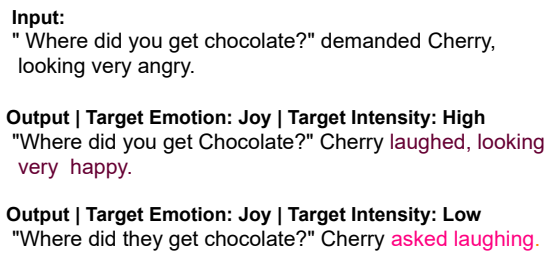

Figure 1: An example to demonstrate the output of our model. The input to the model is a sentence in any emotion, target emotion, and target intensity. The model rewrites the sentence in the target emotion and intensity while preserving the sentence's semantics.

In style-content disentanglement, the model tries to separate the stylistic part of the text from the content of the text either explicitly or in latent representation (implicitly). Different techniques like explicit identification and replacement, back translation, adversarial learning, and controllable generation (Sudhakar et al. (2019), Lee (2020),Shen et al. (2017), Prabhumoye et al. (2018)) are used. However, separating emotions from the content of the text is difficult and, at times, impossible as emotions are intertwined with content. In such a setting style (emotion) - content disentanglement is impossible and unnecessary. So, for the task, we explore style transfer without disentanglement. Techniques such as adversarial learning, controllable generation, reinforcement learning, probabilistic modeling, and pseudo-parallel corpus have been explored in this setting (Lample et al. (2019), Sudhakar et al. (2019), He et al. (2020), Dai et al. (2019), Liu et al. (2021)). We explore reinforcement learning (RL) as it has shown promising results in text style transfer and other NLP tasks. It will also allow us to explore linguistic knowledge of the language model (through rewards) to train our generator, as discussed in the following sections.

Emotion style transfer is challenging as emotions

are on the fence between content and style (Helbig et al., 2020). Unlike sentiment, emotions are gradient in nature; different words/phrases represent different levels of emotional intensity. In this task, we control this gradient nature of the emotion in the output. The aim is to rewrite a given sentence, in any emotion, to a target emotion (say joy) while controlling the intensity of the target emotion (high or low). It is important to note here that input could be any emotion. This setting is different from sentiment or formality style transfer, where we know the style of the source. The target intensity could be either high or low. One approach to solving the problem is first to transfer emotion and then change the intensity of the transferred sentences. However, Goyal et al. (2021) have shown that multiple steps of style transfer suffer from a semantic loss in the output. Thus we approach emotion transfer with required intensity in one single step instead of multiple cascaded steps.

Due to dataset limitations, we only deal with four emotions: anger, fear, joy, and sadness. However, the proposed technique can be easily expanded to other emotions if enough data is available. The key contributions of this paper are:
1) A novel task of emotion transfer with a specified intensity, high or low. Unlike existing style transfer work, the style (emotion) transfer here is gradient in nature, and the input to the model could be from any style (emotion).
2) A novel architecture to solve the task, where the training is bootstrapped by training the model to generate paraphrases. This allows the model to explore lexical and syntactic diversity in generation.
3) Analyzing existing datasets for emotions and intensity, identifying and addressing issues with them. The existing data for emotions is insufficient to train a reinforcement learning based model, so it is augmented with distant learning.

2 Related Work

The reinforcement learning-based approach is a popular technique in unsupervised text style transfer. Xu et al. (2018) explored a cycled reinforcement learning method, Gong et al. (2019) had explored a reinforcement-learning-based generator-evaluator architecture. Luo et al. (2019) proposed a dual reinforcement learning framework. Recent work has used a pre-trained language model's linguistic knowledge and reinforcement learning to achieve a new state-of-the-art unsupervised text style transfer. Liu et al. (2021) has used a GPT 2 (Radford et al., 2019) based model along with direct rewards, Goyal et al. (2021) has used a pre-trained language model to achieve multi-style transfer and Lai et al. (2021) achieved new SOTA in formality style transfer by using linguistic knowledge of BART. Reinforcement learning has also been explored in summarization (Lee and Lee (2017), Paulus et al. (2018)), text simplification (Laban et al., 2021), zero-shot classification (Ye et al., 2020), question generation (Gupta et al., 2020), and neural machine translation (Wu et al., 2018).

Multiple attribute text style transfer is another thread of work related to ours that aims to transfer multiple dimensions of style (say formality, sentiment, gender, etc.). This task was first proposed by Lample et al. (2019). Lample et al. (2019) proposed a novel model that controls several factors of variation in textual data using back-translation. (Syed et al., 2020) explored the linguistic capabilities of a language model to rewrite a text in a target author style. Goyal et al. (2021) explored the rewriting capability of a language model to multiple target-style dimensions by employing multiple style-aware language models as discriminators.

3 Approach

The objective is to rewrite (via a generator g) an input sentence in required target emotion and intensity while preserving the semantics. Let the output be $\hat{x}$, then $\hat{x} = g(x, e, i)$, where x is the source sentence in any emotion. e is the target emotion, and i is the target intensity. i could be either high or low.

3.1 Generator

We use a BART (Lewis et al., 2020) model as a generator g and finetune it to generate $\hat{x}$. BART is a denoising autoencoder for pretraining sequence-to-sequence models. Given the source sentence x and a target sentence $\hat{x}$, the loss function used to finetune the BART model is the cross-entropy between the decoder's output and the target sentence.

3.2 Bootstrapping the Training

We bootstrap the model on a task that helps with our end goal before starting pure reinforcement learning (RL) based training. Existing works use DAE loss (Goyal et al., 2021), cycle consistency loss (Liu et al., 2021), etc., to bootstrap the model. We bootstrap the model by training it to generate paraphrase. As discussed before, emotions are

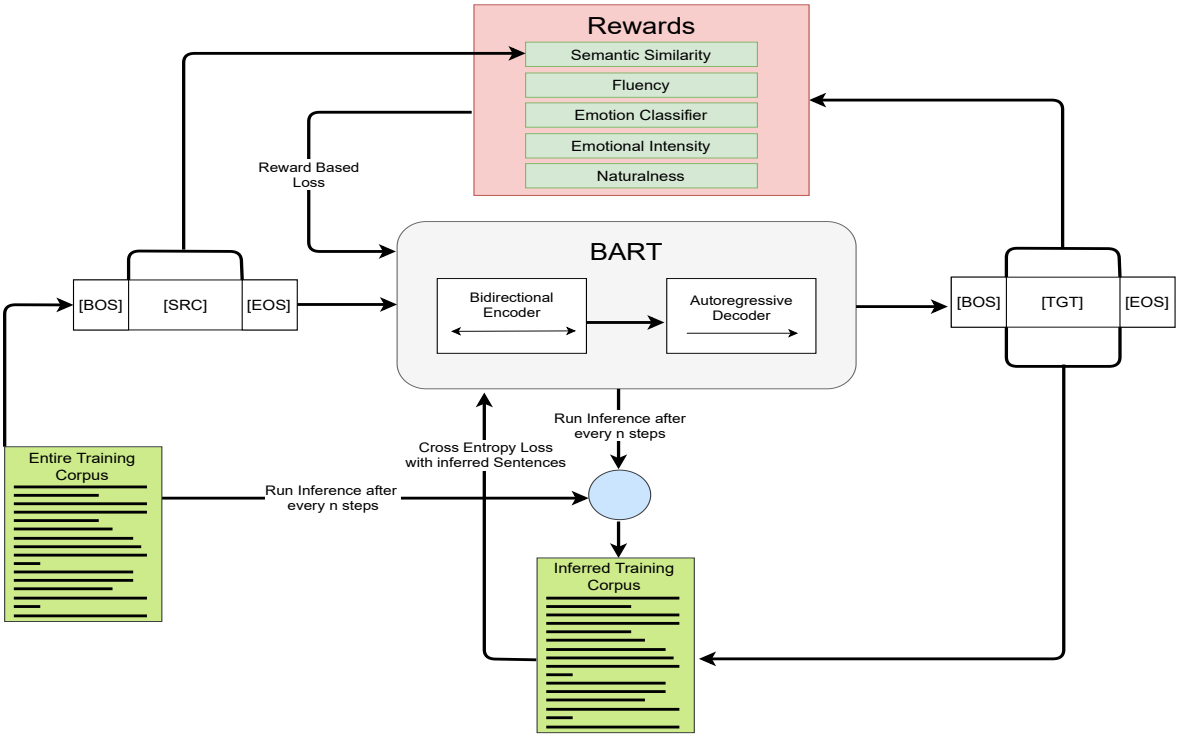

Figure 2: Model Architecture.

gradient in nature, with different words/phrases representing the different intensities of the emotion. So, we need to use different words or phrases based on the target emotion and intensity. Thus having a model that knows how to paraphrase and explore lexical and syntactic diversity helps with our task, as we will show through qualitative and quantitative results. To train the model to generate paraphrases, we use the aggressively filtered subset of PARANMT-50M (Wieting and Gimpel, 2018) released by Krishna et al. (2020). This is a parallel corpus, and the model is trained by minimizing cross-entropy loss between the generator's output and target paraphrase.

$$L(\phi) = -\Sigma i \log(p(\hat{x}_i|\hat{x}_{1:i-1}, x; \phi))$$

Here ϕ denotes the parameters of the generator. In the appendix, we compared our model results when bootstrapped with denoising autoencoding loss and its issues.

3.3 Rewards

Semantic Similarity: This reward ensures that the output of the generator preserves the semantics of the input. To evaluate semantic similarity, existing literature has used the BLEU score (Papineni et al., 2002), however, the BLEU score is restrictive, discourages output diversity, and does not up-weight important semantic words over other words (Krishna et al., 2020). To address this, we use cosine similarity between the sentence embeddings of input and the generator's output. Unlike n-gram metrics, sentence embeddings are not lexically restrictive and will allow the model to explore different words/phrases based on required intensity and emotion. To obtain sentence embeddings we use Sentence-BERT (Reimers and Gurevych, 2019). Thus the semantic similarity rewards (r_{sim}) becomes:

$$r_{sim} = \text{COSINE}(Embedding_{\hat{x}}, Embedding_x)$$

Fluency: This reward ensures that the output sentence is fluent and grammatically correct. We use language model (GPT 2) fluency as described in Laban et al. (2021). The language model assigns a probability to a sequence of words. This probability is used to measure the fluency of generated text. We use the language model to obtain a likelihood of the original paragraph (LM(x)) and of the generated output LM($\hat{x}$). The fluency score (r_{flu}) is given by,

$$r_{flu} = \left[1 - \frac{LM(x) - LM(\hat{x})}{\lambda}\right], \qquad (1)$$

where λ is a tunable hyper-parameter. If the LM($\hat{x}$) is lower than LM(x) by λ or more, r_{flu} = 0. If

Output by generator	TE	TI	r_{sim}	r_{flu}	r_{cla}	r_{int}
While heading to the back of the pub, I glance back at Daweson and wink angry angry.	angry	high	0.839	0.4272	7.6033	1.2304
May I warn you that remaining dormant here could put us in danger happy happy.	joy	low	0.871	0.1461	4.5710	1.4564

Table 1: The table shows the unnatural output that the model produced. The output has a word representing target emotion and intensity added at the end of the sentence. Thus, to prevent our model from producing unnatural output, we use naturalness reward. (TE: Target Emotion and TI: Target Intensity)

LM($\hat{x}$) is above or equal to LM(x), then r_{flu} = 1, and otherwise, it is a linear interpolation. We set λ = 1.3 as it is the value for which the paired Newsela dataset achieves an average LMScore of 0.9.

Emotion Classifier Based Reward: For every target emotion, we finetune a RoBERTa (Liu et al., 2020) based classifier to provide a signal to the generator about the target emotion. This classifier is trained to identify target emotion from other emotions. The log-likelihood of the output being in the target emotion is taken as the classifier-based reward. Thus,

$$r_{cla} = -\log(\ 1 - \text{P(e} \mid \hat{x}, \theta_{cla})),$$

where P(e | $\hat{x}$, θ_{cla}) is the probability of output by generator $\hat{x}$ being in target emotion e and θ_{cla} are parameters of the classifier. RoBERTa model also brings in its linguistic knowledge, thus making the generator more robust.

Emotion Intensity Based Reward: Following existing literature, we define intensity as a real value between 0(low) and 1(high). We finetune a RoBERTa (Liu et al., 2020) based regressor to provide a signal to the generator about the target intensity. This regressor takes a sentence and outputs a real value between $[0, 1]$, which specifies the target emotion's intensity of the sentence. As discussed above, the target intensity could be either high or low. When the target intensity is high, we want the rewards to be large when $\hat{x}$ intensity is close to 1, and if the target intensity is low, we want the rewards to be large when $\hat{x}$ intensity is close to 0. Thus the intensity reward is,

$$r_{int} = \begin{cases} -\log(1 - Reg(\hat{x}, \theta_{reg})), & \text{if } i = high, \\ -\log(Reg(\hat{x}, \theta_{reg})), & \text{if } i = low \end{cases}$$

where Reg($\hat{x}$, θ_{reg}) is the intensity of the generator output $\hat{x}$ as predicted by the RoBERTa based regressor and θ_{reg} are the parameters of the regressor.

Naturalness Reward: On observing the output by the generator when trained on the above rewards, we saw that the generator was producing unnatural sentences by adding words representing target emotion and intensity at the end of the output. As shown in Table 1, these unnatural sentences have high rewards; however, they are undesirable. Liu et al. (2021) also made similar observations and following them, and we train a RoBERTa based classifier to detect if the sentence is natural or not. The log-likelihood of the output being natural is taken as a naturalness-based reward. Thus,

$$r_{nat} = -\log(\ 1 - \text{P(n} \mid \hat{x}, \theta_{nat}))$$

where P(n | $\hat{x}$, θ_{nat}), is the probability of $\hat{x}$ being natural and θ_{nat} is the parameter of naturalness classifier. This classifier is trained along with generator in a GAN like setup. The input to the generator (x) forms positive class and the output of the generator ($\hat{x}$) forms the negative class to train the naturalness classifier. It is trained by minimizing the binary crossentropy loss. We train the naturalness model for 200 steps before using it (i.e. λ_{nat} = 0, if step <=200).

3.4 Learning

All these rewards are discrete sampling, and the gradient could not be propagated through it. Thus, we use REINFORCE (Williams, 1992) algorithm to optimize the model.

$$\nabla_{\theta_g} \mathbb{E}_{\hat{x}\sim p_g(\hat{x})}[r_*(\hat{x})]$$
$$= \mathbb{E}_{\hat{x}\sim p_g(\hat{x})}[\nabla_{\theta_g} \log p_g(\hat{x}) r_*(\hat{x})]$$

Here, r_* is either of the reward discussed in the previous section.

To provide stability to training, we pause the training after every n step. Then we use the generator trained so far to run inference on the training data. Then for the next n step of training, we calculate cross-entropy loss between these inferred sentences and the output of generator g (during training). If we do not update it after every n step, the model is pushed back by the cross-entropy loss. As the training proceeds, the model learns to generate text in the required style, but the cross-entropy loss is calculated with outdated sentences, thus pushing the model back. We use $n = 8000$. Thus,

$$L_{ce}(\phi) = -\Sigma i \log(p(\hat{x}_i|\hat{x}_{1:i-1}, x'; \phi))$$

where x' represents inferred sentence and ϕ denotes the parameters of the generator.

To train the generator, we use the weighted average of the losses defined above:

$L(\theta_g) = \lambda_{ce}L_{ce}(\theta_g) + \lambda_{sim}L_{sim}(\theta_g) + \lambda_{flu}L_{flu}(\theta_g) + \lambda_{cla}L_{cla}(\theta_g) + \lambda_{int}L_{int}(\theta_g) + \lambda_{nat}L_{nat}(\theta_g)$

where λ denotes the weight of the corresponding term. Optimal value of different λ is reached after extensive experimentation. We observed that linearly decreasing λ_{cla} and linearly increasing λ_{int} produced better results. Figure 2 shows the overall architecture of the model.

4 Dataset & Training

Emotion Intensity: We use the dataset released by Mohammad et al. (2018) to train our **intensity** model. It is the largest emotion intensity dataset and is created through crowdsourcing. This dataset consists of a tweet and its emotional intensity and covers four emotions, anger, fear, joy, and sadness. The emotional intensity is a real value between 0 (low) and 1 (high). Since this dataset consists of tweets, one needs to be careful in using it as it is abundant in emoji and hashtags. Emojis and hashtags (at the end) are generally used to express emotions and could affect the intensity of the tweet. Thus, we remove all the tweets that have emojis or hashtags at the end. This ensures that the model learns intensity through the content of the tweet and not emojis and hashtags.

The username in Twitter are fancy consisting of a combination of numbers, names, underscore, etc. We observed that these fancy tweet handles were affecting the performance of the model. So, we tried three different approaches to deal with it:

1. Replace tweet handles with a special token ([NAME]).
2. Completely removing tweet handles.
3. Replace tweet handles with a random name. We use the names library[1] for this.

We tested on tweets that do not contain any username and found that replacing tweet handles with a random name performed the best. Thus, we replace tweet handles with a random name. We trained a RoBERTa based regression model for all our experiments due to RoBERTa's excellent performance in various NLP tasks and the linguistic knowledge it carries. Following Mohammad et al. (2018) we use the Pearson Correlation Coefficient to evaluate our models. Table 2 contains the dataset statistics and the Pearson Correlation Coefficient of our model on the test set.

[1] https://pypi.org/project/names/

Emotion	Train	Test	PCC
Anger	1102	515	0.811
Fear	1243	498	0.777
Joy	967	323	0.812
Sadness	1001	399	0.790

Table 2: Intensity dataset statistics and the Pearson Correlation Coefficient on the test set. PCC: Pearson Correlation Coefficient

Emotion	Train	Test	Acc.	M F1
Anger	1400	246	93.74%	0.889
Fear	1200	277	93.14%	0.889
Joy	1700	300	97.07%	0.954
Sadness	1300	308	93.08%	0.892
No Emo.	1400	371	-	-

Table 3: Emotion classification dataset statistics and results on the test set. M F1: Macro F1 Score, No Emo.: No emotions

Emotion Classification: Several datasets have been created and published for emotion classification. Bostan and Klinger (2018) studied fourteen different datasets for emotion understanding and combined them into a single large dataset. Having data from multiple sources is good as the model becomes more robust. However, some of these fourteen datasets have their limitations. Strapparava and Mihalcea (2007) consists of news headlines and thus are not complete sentences. Also, it is a multi-label annotated dataset and thus cannot be used for our purpose. Crowdflower 2016, Mohammad et al. (2014), Liu et al. (2017), Schuff et al. (2017), and Mohammad (2012) are the datasets containing text from social media, and thus sentences are very ill-formed, ungrammatical, have heavy dependencies on hashtags and emojis, and have noisy labeling. Therefore after careful consideration, we use the following datasets.

1. Blogs dataset released by Aman and Szpakowicz (2007).
2. Emotion stimulus dataset released by Ghazi et al. (2015). It consists of sentences annotated with the cause of the emotion.
3. Dialouges dataset released by Li et al. (2017). It consists of sentences from conversations.
4. Tales corpus released by Alm and Sproat (2005). It consists of sentences from fairytales.
5. Dataset used in training the intensity model released by Mohammad et al. (2018). Since it is the

Emotion	Train	Test	Acc.	M F1
Anger	1400	246	93.74%	0.889
Fear	1200	277	93.14%	0.889
Joy	1700	300	97.07%	0.954
Sadness	1300	308	93.08%	0.892
No Emo.	1400	371	-	-

Table 4: Emotion classification dataset statistics and results on the test set. M F1: Macro F1 Score, No Emo.: No emotions

Emotion	Train	Test
Anger	36000	1880
Fear	36000	1520
Joy	36000	1872
Sadness	36000	1162

Table 5: Distantly learned data for RL training

largest among the five datasets, we use a subset of it to prevent the model from learning its biases.

We finetune a RoBERTa model to identify target emotion from other emotions. We report classification accuracy, Macro F1 score, and statistics for each emotion in Table 4. No emotions data is collected from the Blogs dataset. Note that the statistics in the table represent the number of sentences in that emotion. The classifier was trained to differentiate the target emotion (labeled 1) from others (labeled 0).

Reinforcement Learning Training: For RL training, we need a large amount of data. However, the data described in section 4.2 is comparatively smaller and insufficient for RL training. So we augment this data through distant supervision. Specifically, we take the classifier trained in section 4.2 and use it to identify emotional sentences in the book corpus [2] (classification probability >= 0.9). Table 5 shows the statistics of the number of sentences in different emotions collected through distant learning.

Experimental Details: We train the model for 75,000 steps on two RTX 2080 TI (11 GB). It takes about 28 hours for the entire training (including bootstrapping). We use a learning rate of 10^{-5} and a batch size of 6. As discussed we ran inference on training data after every 8000 step.

Table 6 shows different weights of the loss terms. As discussed, we found that linearly decreasing the weight of the classifier loss and linearly increasing the weight of the intensity loss produced better results than keeping them constant. It must be noted that some other values of these weights might perform better on automatic metrics, but the output quality was poor. So, we manually checked the outputs to arrive at the most appropriate weights.

[2] https://battle.shawwn.com/sdb/books1/books1.tar.gz

5 Results and Discussions

We evaluate the performance of our model on four grounds: 1) Semantic similarity between input and output. This is measured using cosine similarity between sentence embeddings of input and output. 2) Language model perplexity to measure fluency and grammatical correctness of the output. It should be low. 3) Transfer Accuracy, measured through the emotion classifier that we trained above. 4) Output intensity, measured through the intensity regressor we trained above. We test the performance of our model on two different test datasets - created through human annotation (described in section 4.2) and learned through distant supervision (described in section 4.3).

5.1 Automatic Evaluations

Table 7 shows the results of automatic evaluations on Human annotated data. Please refer to appendix for results on distantly learned data.

The transfer strength is low when the target intensity is low. This is expected because once we go to lower intensity, sometimes the classifier is not able to detect the transformed sentence as being in target emotion. Higher intensity sentences are easier to detect by a classifier compared to a lower intensity sentence.

While going to lower intensity, we must be careful. If the model does not change a few sentences, then the intensity of such output would be low, and thus overall average intensity would be low. This will be a false positive, as we do not know if the model is going for lower intensity or is it just not changing a few sentences and getting a low-intensity score. So we study the cosine scores, intensity, and perplexity of only those sentences identified to be in target emotion by our classifier. This study suggests that the model is writing sentences in target intensity with high cosine similarity and low perplexity. Please refer to appendix for exact scores.

The trend suggests that semantic similarity is easy to achieve when the target intensity is low, but at the same time, the fluency of the model suffers when going for lower intensity.

TE	TI	λ_{ce}	λ_{sim}	λ_{flu}	$\lambda_{cla\ start}$	$\lambda_{cla\ end}$	$\lambda_{int\ start}$	$\lambda_{int\ end}$	λ_{nat}
Anger	high	0.75	0.75	0.75	1.5	0.1	1	2.5	1
	low	0.75	0.75	0.75	1.5	0.5	0.1	2	1
Fear	high	0.75	0.75	0.75	1	0.5	0.1	1.5	1
	low	0.75	0.75	0.75	1.5	0.5	0.1	2.5	1
Joy	high	0.75	0.75	0.75	1.5	0.1	1	2.5	1
	low	0.75	0.75	0.75	1.5	0.5	0.1	2	1
Sadness	high	0.75	0.75	0.75	1.5	0.1	1	2.5	1
	low	0.75	0.75	0.75	2	0.5	0.1	3	1

Table 6: Weights for different loss terms. TE: Target Emotion, TI: Target Intensity

Model	TE	TI	Semantic Similarity	Perplexity	Classification Accuracy	Intensity
Our	Anger	High	0.780	188.54	64.58	0.571
Our		Low	0.801	189.81	46.79	0.423
Our	Fear	High	0.750	163.59	55.62	0.600
Our		Low	0.812	229.73	54.26	0.375
Our	Joy	High	0.729	164.77	77.80	0.491
Our		Low	0.762	193.63	63.37	0.335
Our	Sadness	High	0.749	187.69	71.11	0.565
Our		Low	0.762	164.02	59.83	0.444

Table 7: Automated evaluations results on human annotated test data. TE: Target Emotion, TI: Target Intensity.

TE	TI	MS	GC	TS	Intensity
Anger	High	3.96	4.32	4.49	83.12
	Low	4.03	4.18	3.55	63.23
Fear	High	4.12	3.93	4.19	73.38
	Low	3.74	3.83	3.16	51.77
Joy	High	3.72	4.25	4.25	73.45
	Low	3.67	4.02	3.63	58.27
Sadness	High	3.73	3.69	3.90	72.02
	Low	3.51	3.71	2.90	45.69

Table 8: Human evaluations results. TE: Target Emotion, TI: Target Intensity (integer value between 0 and 100), MS: Meaning Simiarity, GC: Grammatical Correctness, TS: Transfer Strength.

5.2 Human Evaluations

While automatic metrics provide an understanding of how good our model is, they have their limitations. So to better understand the performance of the model, we conducted extensive human evaluations. We evaluated the model on four parameters - meaning similarity, transfer strength, grammatical correctness, and target emotion intensity. The first three parameters were measured on a five-point Likert scale. Participants had to give an integer value between 0 (low intensity) and 100 (high intensity) for target emotion intensity. Participants were shown the input sentence and two possible outputs for given target emotion (one output for high intensity and the other for low) in a randomized order. Following the guidelines by Clark et al. (2021), we provide ample examples to the participants for them to judge more accurately.

We took 32 triplets (input, high-intensity output, and low-intensity output), with eight triplets from each target emotion. Each triplet was annotated by atleast 3 participants. Table 8 shows the results of human evaluations. These results are in line with our automatic evaluations and thus add confidence to the efficacy of our proposed approach in achieving emotion style transfer with a specified intensity.

5.3 Qualitative Examples

Table 9 shows some output using the approach. The first output offers an interesting insight into the model. While the phrase *shouts angrily* indicates that the target emotion is anger, the intensity is controlled by the sentence's first part. The model changed *Fucking right, I'll!* to *I'll take care of you!* for the low-intensity output. We observed in our dataset that the word *Fucking* usually occurs in a high-intensity setting, and thus the model takes care of it. The second depicts how bootstrapping using paraphrasing helped. The paraphrase's knowledge allowed the model to start the sentence with *What a beautiful....* for high-intensity output and thus it expored syntactic diversity. Starting the sentence with *What* is stressing on *beauty* and how it is a big cause of *worry* (notice the use of the word *concerned* in low-intensity output). Third example shows how our model uses punctuation to express intensity. Notice the use of an exclamation mark in high-intensity output and a question mark in low-intensity output. The last output shows varied intensity output for target emotion as sadness.

	TE	TI	Sentence
Input			"Fucking right I will!" Bill shouts nervously.
Output	Anger	High	"Fucking right, I'll!" Bill shouts angrily.
Output	Anger	Low	"I'll take care of you!" Bill shouts angrily.
Input			The young lord was overjoyed to see what a beautiful wife his friends had found for him.
Output	Fear	High	What a beautiful wife his friends found for him worried the young lord.
Output	Fear	Low	The young lord was concerned to see what a beautiful wife his friends had found for him.
Input			"I want to know where Frank is!" Sparky could hardly believe he was yelling.
Output	Joy	High	"I want to know where Frank is!" Sparky laughed, surprised he was happy.
Output	Joy	Low	"I want to know where Frank is?" Sparky asked laughing.
Input			"Richard?" she asked, fear making her blood run cold.
Output	Sadness	High	"Richard?" She asked, her blood sank with despair.
Output	Sadness	Low	"Richard?" She frowned, her blood heat up.

Table 9: Some outputs of our model. TE: Target Emotion, TI: Target Intensity

6 Conclusion

This work proposed and solved a novel problem statement of emotional style transfer with a specified intensity. The proposed BART-based deep reinforcement learning-based architecture can rewrite an input sentence in required intensity and target. Qualitative and quantitative results show that bootstrapping the model by training it to generate paraphrases helped the model explore various lexicons based on the need. Through extensive human and automatic evaluations, we show the efficacy of our model. Our code and associated dataset will be made open source.

References

Cecilia Ovesdotter Alm and Richard Sproat. 2005. Emotional sequencing and development in fairy tales. In *Affective Computing and Intelligent Interaction*, pages 668–674, Berlin, Heidelberg. Springer Berlin Heidelberg.

Saima Aman and Stan Szpakowicz. 2007. Identifying expressions of emotion in text. pages 196–205.

Laura Ana Maria Bostan and Roman Klinger. 2018. An analysis of annotated corpora for emotion classification in text. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2104–2119. Association for Computational Linguistics.

Keith Carlson, Allen Riddell, and Daniel N. Rockmore. 2017. Zero-shot style transfer in text using recurrent neural networks. *CoRR*, abs/1711.04731.

Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A. Smith. 2021. All that's 'human' is not gold: Evaluating human evaluation of generated text. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7282–7296, Online. Association for Computational Linguistics.

Ning Dai, Jianze Liang, Xipeng Qiu, and Xuanjing Huang. 2019. Style transformer: Unpaired text style transfer without disentangled latent representation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5997–6007, Florence, Italy. Association for Computational Linguistics.

Diman Ghazi, Diana Inkpen, and Stan Szpakowicz. 2015. Detecting emotion stimuli in emotion-bearing sentences. In *Computational Linguistics and Intelligent Text Processing*, pages 152–165, Cham. Springer International Publishing.

Hongyu Gong, Suma Bhat, Lingfei Wu, JinJun Xiong, and Wen-mei Hwu. 2019. Reinforcement learning based text style transfer without parallel training corpus. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3168–3180, Minneapolis, Minnesota. Association for Computational Linguistics.

Navita Goyal, Balaji Vasan Srinivasan, Anandhavelu N, and Abhilasha Sancheti. 2021. Multi-style transfer with discriminative feedback on disjoint corpus. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3500–3510, Online. Association for Computational Linguistics.

Deepak Gupta, Hardik Chauhan, Ravi Tej Akella, Asif Ekbal, and Pushpak Bhattacharyya. 2020. Reinforced multi-task approach for multi-hop question generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2760–2775, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Junxian He, Xinyi Wang, Graham Neubig, and Taylor Berg-Kirkpatrick. 2020. A probabilistic formulation of unsupervised text style transfer. *CoRR*, abs/2002.03912.

David Helbig, Enrica Troiano, and Roman Klinger. 2020. Challenges in emotion style transfer: An exploration with a lexical substitution pipeline. *CoRR*, abs/2005.07617.

Zhiqiang Hu, Roy Ka-Wei Lee, and Charu C. Aggarwal. 2020. Text style transfer: A review and experiment evaluation. *CoRR*, abs/2010.12742.

Harsh Jhamtani, Varun Gangal, Eduard Hovy, and Eric Nyberg. 2017. Shakespearizing modern language using copy-enriched sequence to sequence models. In *Proceedings of the Workshop on Stylistic Variation*, pages 10–19, Copenhagen, Denmark. Association for Computational Linguistics.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 737–762, Online. Association for Computational Linguistics.

Philippe Laban, Tobias Schnabel, Paul Bennett, and Marti A. Hearst. 2021. Keep it simple: Unsupervised simplification of multi-paragraph text. *CoRR*, abs/2107.03444.

Huiyuan Lai, Antonio Toral, and Malvina Nissim. 2021. Thank you BART! rewarding pre-trained models improves formality style transfer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 484–494, Online. Association for Computational Linguistics.

Guillaume Lample, Sandeep Subramanian, Eric Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2019. Multiple-attribute text rewriting. In *International Conference on Learning Representations*.

Gyoung Ho Lee and Kong Joo Lee. 2017. Automatic text summarization using reinforcement learning with embedding features. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 193–197, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Joosung Lee. 2020. Stable style transformer: Delete and generate approach with encoder-decoder for text style transfer. *CoRR*, abs/2005.12086.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. DailyDialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 986–995, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Vicki Liu, Carmen Banea, and Rada Mihalcea. 2017. Grounded emotions. pages 477–483.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Ro{bert}a: A robustly optimized {bert} pretraining approach.

Yixin Liu, Graham Neubig, and John Wieting. 2021. On learning text style transfer with direct rewards. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4262–4273, Online. Association for Computational Linguistics.

Fuli Luo, Peng Li, Jie Zhou, Pengcheng Yang, Baobao Chang, Xu Sun, and Zhifang Sui. 2019. A dual reinforcement learning framework for unsupervised text style transfer. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5116–5122. International Joint Conferences on Artificial Intelligence Organization.

Aman Madaan, Amrith Setlur, Tanmay Parekh, Barnabas Poczos, Graham Neubig, Yiming Yang, Ruslan Salakhutdinov, Alan W Black, and Shrimai Prabhumoye. 2020. Politeness transfer: A tag and generate approach. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1869–1881, Online. Association for Computational Linguistics.

Saif Mohammad. 2012. #emotional tweets. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 246–255, Montréal, Canada. Association for Computational Linguistics.

Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. SemEval-2018 task 1: Affect in tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 1–17, New Orleans, Louisiana. Association for Computational Linguistics.

Saif Mohammad, Xiaodan Zhu, Svetlana Kiritchenko, and Joel Martin. 2014. Sentiment, emotion, purpose, and style in electoral tweets. 51.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*.

Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 866–876, Melbourne, Australia. Association for Computational Linguistics.

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

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may I introduce the GYAFC dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 129–140, New Orleans, Louisiana. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Hendrik Schuff, Jeremy Barnes, Julian Mohme, Sebastian Padó, and Roman Klinger. 2017. Annotation, modelling and analysis of fine-grained emotions on a stance and sentiment detection corpus. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 13–23, Copenhagen, Denmark. Association for Computational Linguistics.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6833–6844, Red Hook, NY, USA. Curran Associates Inc.

Carlo Strapparava and Rada Mihalcea. 2007. SemEval-2007 task 14: Affective text. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 70–74, Prague, Czech Republic. Association for Computational Linguistics.

Akhilesh Sudhakar, Bhargav Upadhyay, and Arjun Maheswaran. 2019. "transforming" delete, retrieve, generate approach for controlled text style transfer. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3269–3279, Hong Kong, China. Association for Computational Linguistics.

Bakhtiyar Syed, Gaurav Verma, Balaji Vasan Srinivasan, Anandhavelu Natarajan, and Vasudeva Varma. 2020. Adapting language models for non-parallel author-stylized rewriting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9008–9015.

John Wieting and Kevin Gimpel. 2018. ParaNMT-50M: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Melbourne, Australia. Association for Computational Linguistics.

Ronald J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3–4):229–256.

Lijun Wu, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2018. A study of reinforcement learning for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3612–3621, Brussels, Belgium. Association for Computational Linguistics.

Jingjing Xu, Xu Sun, Qi Zeng, Xiaodong Zhang, Xuancheng Ren, Houfeng Wang, and Wenjie Li. 2018. Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 979–988, Melbourne, Australia. Association for Computational Linguistics.

Zhiquan Ye, Yuxia Geng, Jiaoyan Chen, Jingmin Chen, Xiaoxiao Xu, SuHang Zheng, Feng Wang, Jun Zhang, and Huajun Chen. 2020. Zero-shot text classification via reinforced self-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3014–3024, Online. Association for Computational Linguistics.

A Appendix

The appendix includes:
1) Results of automatic evaluation of when tested on distantly learned data.
2) Results on output sentences identified to be in target emotion.
3) Results when bootstrapping is done with DAE loss instead of paraphrasing.
4) More outputs by our model.

B Results of distant learned data

Table 10 shows the results of automatic evaluations of our model on data created through distant supervision as described in section 4.3. The numbers, in general, are better when compared to human-created data. This is expected as automatically created data carries the model's biases, which the generator could exploit. Human-created data shows much more diversity and, in general, more challenging to deal with than distantly learned data.

C Results on output sentences identified to be in target emotion

As discussed in section 6.1, we need to be careful while evaluating the model, especially when target intensity is low. If the model does not change a few sentences, then the intensity of such output would be low, and thus overall average intensity would be low. This will be a false positive, as we do not know if the model is going for lower intensity or is it just not changing a few sentences and getting a low-intensity score. So, we study the cosine scores, intensity, and perplexity of only those sentences identified to be in target emotion by our classifier. Table 11 shows the results for the sentences in the target emotions. It suggests that the model is writing sentences in required intensity with high cosine similarity and a little high perplexity.

D Results when bootstrapping is done with DAE loss instead of paraphrasing.

We have bootstrapped our model by training it to generate paraphrases. This will allow greater lexical and syntactic diversity needed for target emotion and intensity. Another popular approach to bootstrap the training is to train the model by optimizing on denoising autoencoding loss. The model is given a noisy version of the text, and it is trained to reconstruct the sentence. The noisy version is created by replacing a token with a special mask (<mask>) token with a probability of 0.15.

Table 12 compares the performance of our model when bootstrapped with DAE loss. We see that the model performs well in semantic similarity, transfer accuracy, and intensity metrics but has very poor perplexity. On closely examining the output of the DAE bootstrapped model, we observed that the model is replacing random words in the input sentence with words that represent target emotion and intensity, which is undesirable. This is the reason semantic similarity is high as only a few words are getting replace. Table 13 shows some such outputs. Emotion and its intensity are a function of a sentence and not just random words, and thus, replacing random words will not solve our task. Thus, though the DAE bootstrapped model results seem better in numbers, the actual output sentences are wrong, bad, and undesirable.

E More outputs by our model

Table 14 shows some more outputs by our model for better qualitative understanding of our approach and model.

Model	TE	TI	SS	Perplexity	CA	Intensity
Our	Anger	High	0.783	147.36	80.40	0.559
Our		Low	0.801	134.57	67.36	0.444
Our	Fear	High	0.753	137.16	70.01	0.590
Our		Low	0.790	184.51	69.72	0.357
Our	Joy	High	0.751	145.65	81.77	0.492
Our		Low	0.768	159.61	76.09	0.364
Our	Sadness	High	0.767	136.46	78.24	0.539
Our		Low	0.765	145.56	76.02	0.457

Table 10: Automated evaluations results on distantly learned test data. TE: Target Emotion, TI: Target Intensity, SS: Semantic Similarity, CA: Classification Accuracy.

TE	TI	Semantic Similarity	Perplexity	Intensity
Human Annotated Data				
Anger	High	0.768	208.68	0.613
	Low	0.797	263.36	0.513
Fear	High	0.715	148.24	0.715
	Low	0.806	239.81	0.442
Joy	High	0.706	157.74	0.527
	Low	0.754	207.23	0.381
Sadness	High	0.706	209.76	0.593
	Low	0.752	189.23	0.501
Distantly Learned Data				
Anger	High	0.771	145.29	0.585
	Low	0.797	147.60	0.498
Fear	High	0.732	129.80	0.963
	Low	0.784	183.33	0.384
Joy	High	0.730	142.14	0.519
	Low	0.763	162.40	0.393
Sadness	High	0.711	124.83	0.572
	Low	0.755	148.19	0.487

Table 11: Automated evaluations results on sentences identified to be in target emotion. TE: Target Emotion, TI: Target Intensity.

Model	TE	TI	SS	Perplexity	CA	Intensity
Human Annotated Data						
DAE	Anger	High	0.766	491.71	44.67	0.482
Paraphrase			0.780	188.54	64.58	0.571
DAE		Low	0.842	346.73	15.33	0.391
Paraphrase			0.801	189.81	46.79	0.423
DAE	Fear	High	0.657	332.73	91.58	0.812
Paraphrase			0.750	163.59	55.62	0.600
DAE		Low	0.853	355.90	49.82	0.462
Paraphrase			0.812	229.73	54.26	0.375
DAE	Joy	High	0.846	428.38	53.47	0.364
Paraphrase			0.729	164.77	77.80	0.491
DAE		Low	0.835	379.18	62.35	0.332
Paraphrase			0.762	193.63	63.37	0.335
DAE	Sadness	High	0.828	449.79	87.98	0.657
Paraphrase			0.749	187.69	71.11	0.565
DAE		Low	0.813	440.13	86.21	0.541
Paraphrase			0.762	164.02	59.83	0.444
Distantly Learned Data						
DAE	Anger	High	0.781	174.26	57.11	0.469
Paraphrase			0.783	147.36	80.40	0.559
DAE		Low	0.847	165.32	26.02	0.376
Paraphrase			0.801	134.57	67.36	0.444
DAE	Fear	High	0.747	241.87	97.67	0.770
Paraphrase			0.753	137.16	70.01	0.590
DAE		Low	0.856	389.45	71.46	0.473
Paraphrase			0.790	184.51	69.72	0.357
DAE	Joy	High	0.859	389.63	69.87	0.397
Paraphrase			0.751	145.65	81.77	0.492
DAE		Low	0.846	208.15	83.68	0.363
Paraphrase			0.768	159.61	76.09	0.364
DAE	Sadness	High	0.786	306.77	97.32	0.679
Paraphrase			0.767	136.46	78.24	0.539
DAE		Low	0.828	289.78	92.64	0.508
Paraphrase			0.765	145.56	76.02	0.457

Table 12: Automated evaluations results when bootstrapping is done with DAE loss. TE: Target Emotion, TI: Target Intensity, SS: Semantic Similarity, CA: Classification Accuracy.

	TE	TI	Sentence
Input			Oh, I didn't know, perhaps I shouldn't ask him to come then?
Output	Sadness	High	Oh, I didn't sad, perhaps I shouldn't ask him to come then?
Input			But what about poor Gussie, look at the state he's in!
Output	Joy	Low	But what about poor Gussie, look at the state he's laughing.
Input			Then anger sparked through me. "What the fuck for?"
Output	Fear	High	Then anger sparked through me frightened. "What the fuck for?"

Table 13: Examples depicting issues with DAE loss. Random words are getting replaced with words denoting target emotion and intensity. Such sentences are grammatically incorrect and do not solve our goal to rewrite a sentence in given target emotion and intensity.

	TE	TI	Sentence
Input			Made a right tit of yourself last night, didn't you, Simes?
Output	Anger	High	You've fucking made a right tit of yourself, haven't you, Simes?
Output	Anger	Low	You made a right tit out of yourself last night, didn't you?
Input			Taysha couldn't help but smile at the men's teasing.
Output	Anger	High	Taysha couldn't contain her anger with the men's teasing.
Output	Anger	Low	Taysha smiled angrily at the men's teasing.
Input			"Owww!" Anastasia clutched my hair, forcing me to lean over her.
Output	Anger	High	Anastasia clutched my hair angrily, forced me to sit down next to her.
Output	Anger	Low	"oh, my God!" Anastasia clutched my hair and caused me to lean
Input			Mizou told me. Do you need to sit down? You look pretty shaken.
Output	Anger	High	You look very angry. Do you have to sit down? Mizou asked angrily.
Output	Anger	Low	Mizou told angrily, "do you have to sit down? You look scared.
Input			"How the hell do we carry this a mile down the road?" asked Forest.
Output	Fear	High	"how the hell do we carry it a mile?" Forest asked afraidly.
Output	Fear	Low	"how do we carry it a mile down the road?" Forest asked concernedly.
Input			His parents were furious over the engagement business
Output	Fear	High	The engagement business scared his parents.
Output	Fear	Low	The engagement business was concerned by his parents.
Input			A chill shot down Joe's spine. "You sure did do good, Marie. You sure did."
Output	Fear	High	"you've done well, Marie," Joe said with a worried feeling.
Output	Fear	Low	"you really did good, Marie," Joe said concernedly.
Input			" Debbie would be pleased , " she said .
Output	Fear	High	" Debbie would be happy," she said afraidly.
Output	Fear	Low	" Debbie'd like, " she said concernedly.
Input			"You brought a cat through a portal!!" he exclaimed.
Output	Joy	High	"you brought a cat through the portal!" He laughed, happy.
Output	Joy	Low	"you brought a cat through the portal?" He asked laughing.
Input			Yes, I was annoyed she hadn't told me back then.
Output	Joy	High	She didn't tell me back then. I was happy.
Output	Joy	Low	Yes, I was surprised she didn't tell me.
Input			Yep," Dad replied with a chuckle. "You were as nervous as an ant on a hot tin can.
Output	Joy	High	"you were happy, like an ant on a hot tin can," Dad laughed.
Output	Joy	Low	"yes," Dad said, "you were happy as an ant on a hot tin can.
Input			"But how do you know Redfeld?" insisted Tom, still alarmed at this revelation.
Output	Joy	High	"how do you know Redfeld?" Tom laughed, still happy at the revelation.
Output	Joy	Low	"but how do you know Redfeld?" Tom asked joyfully.
Input			Then I forgot all about my worries and doubts, and my adventure began.
Output	Sadness	High	I forgot all about my worries and doubts, and I started grimly.
Output	Sadness	Low	Then I forgot my despair and doubts, and my adventure began.
Input			She gazed at Ruth , and her face seemed to fill up with horror.
Output	Sadness	High	She looked despairing at Ruth and her face filled with despair.
Output	Sadness	Low	She frowned at Ruth, and her face widened.
Input			Pigling Bland, much alarmed, determined to leave at daybreak.
Output	Sadness	High	Pigling Bland, much despairing and wanting to go at daybreak.
Output	Sadness	Low	Pigling Bland frowned much, ready to leave at daybreak.
Input			And I couldn't even dream last night, so now I'm deprived of that.
Output	Sadness	High	I couldn't dream last night, so now I'm down grimly.
Output	Sadness	Low	I couldn't dream last night, so now I'm missing it.

Table 14: Model's Output