Reinforced Counterfactual Data Augmentation for Dual Sentiment Classification

Hao Chen, Rui Xia[*] and Jianfei Yu
School of Computer Science and Engineering,
Nanjing University of Science and Technology, China
{hchen, rxia, jfyu}@njust.edu.cn

Abstract

Data augmentation and adversarial perturbation approaches have recently achieved promising results in solving the over-fitting problem in many natural language processing (NLP) tasks including sentiment classification. However, existing studies aimed to improve the generalization ability by augmenting the training data with synonymous examples or adding random noises to word embeddings, which cannot address the spurious association problem. In this work, we propose an end-to-end reinforcement learning framework, which jointly performs counterfactual data generation and dual sentiment classification. Our approach has three characteristics: 1) the generator automatically generates massive and diverse antonymous sentences; 2) the discriminator contains a original-side sentiment predictor and an antonymous-side sentiment predictor, which jointly evaluate the quality of the generated sample and help the generator iteratively generate higher-quality antonymous samples; 3) the discriminator is directly used as the final sentiment classifier without the need to build an extra one. Extensive experiments show that our approach outperforms strong data augmentation baselines on several benchmark sentiment classification datasets. Further analysis confirms our approach's advantages in generating more diverse training samples and solving the spurious association problem in sentiment classification.

1 Introduction

Deep learning techniques (e.g., CNN, RNN, pretrained language models) have achieved great success in many natural language processing (NLP) tasks including sentiment classification. Despite their promising results, recent studies reported that due to the over-fitting problem these models may easily fail in attacking examples with even little modification on real examples (Iyyer et al., 2018; Ren et al., 2019; Zhang et al., 2020; Xing et al., 2020). Researchers have attempted to address this issue from two main perspectives: data augmentation and adversarial perturbation. The former tries to augment the training data by generating synonymous sentences (Zhang et al., 2015; Kobayashi, 2018; Xu et al., 2019); the latter aims to improve the generalization ability by applying perturbations to the word embeddings (Miyato et al., 2017; Croce et al., 2020). Although these methods have achieved sound performance, they still suffer from the *spurious association* problem. Machine learning systems are trained to exploit the associations between the input features and the output labels to make accurate predictions. For example, if a neutral word (e.g.,"*book*") occurs more frequently in the positive class than in the negative class of the training data, "*book*" will have a *spurious association* with the positive class.

Recently, counterfactual data augmentation has shown to be an effective way to address the *spurious association* problem in sentiment classification (Kaushik et al., 2020; Wang and Culotta, 2021; Xing et al., 2020; Xia et al., 2013, 2015b). The core idea behind this line of work is to construct training and test samples by generating antonymous sentences and reversing its sentiment label. In the previous example, by generating an antonymous sample for each training sample, the frequency of "*book*" in the negative class will also increase, and thus the *spurious association* between "*book*" and the positive class will be alleviated.

However, these methods still have three shortcomings: 1) They either relied on human efforts or resorted to rules for antonymous sample construction which is labor-intensive and timecosting. The diversity of generated samples is also limited; 2) They regarded antonymous sample generation and sentiment classification as two separate tasks, and pipeline them; 3) They mostly

[*] Corresponding author.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 269–278
November 7–11, 2021.

merged the generated antonymous samples into the original training set, and ignored the one-to-one correspondence between the antonymous and original samples.

In this paper, we propose an end-to-end reinforcement learning framework named Reinforced Counterfactual Data Augmentation (RCDA) for joint counterfactual data augmentation and dual sentiment classification. The counterfactual sentence generation and the dual sentiment classification modules are regarded as a generator and a discriminator, and integrated in a reinforcement learning framework. First, the generator uses one-to-many antonym and synonym lists obtained from WordNet to generate massive antonymous candidates based on multi-label learning, and automatically select the best antonymous sentence based on reinforcement learning. Second, the discriminator contains an original-side sentiment predictor and an antonymous-side sentiment predictor, which regards the original and antonymous samples as pairs to perform dual sentiment classification. The action reward in reinforcement learning is also computed based on both original and antonymous sides. Finally, the discriminator can be directly used as the final sentiment classifier for the testing examples.

Extensive experiments on four benchmark datasets indicate that our approach significantly outperform strong data augmentation baselines. Further analysis demonstrates that our method is more effective in generating diverse training samples and alleviating the spurious association problem in sentiment classification.

The contributions of this paper can be summarized as follows:

- We propose a new framework for joint counterfactual data generation and dual sentiment classification.[1]
- We generate many antonymous candidates for each original sample and select the best one, which improves the quality and diversity of the generated samples.
- We regard the antonymous and original samples as pairs, and feed them to the discriminator for dual training and prediction, which alleviates the spurious association problem in sentiment classification.

2 Related Work

With the recent advances of deep learning (Socher et al., 2013; Kim, 2014; Tai et al., 2015; Joulin et al., 2017; Johnson and Zhang, 2017; Devlin et al., 2019), the performance of sentiment classification has been significantly improved. However, these models were typically data-driven and lack of generalization ability. Some previous studies pointed out that adding a slight disturbance to the test data may lead to incorrect predictions (Iyyer et al., 2018; Ren et al., 2019; Zhang et al., 2020; Xing et al., 2020).

The studies that attempted to improve the generalization ability of neural network models in NLP can be roughly divided into three categories.

Adding perturbation focused on applying perturbations to the word embeddings (Miyato et al., 2017; Croce et al., 2020), adding regularization terms, or using the dropout strategy (Hinton et al., 2012).

Synonymous sample generation aimed to randomly replace some words in the real samples with their synonyms, hypernyms, or hyponyms from WordNet to generate a large amount of synonymous samples (Zhang et al., 2015; Kobayashi, 2018; Xu et al., 2019). However, these methods tend to suffer from the *spurious association* problem. It is worth noting that our model is similar to Xu et al. (2019), but there are a number of major differences. Firstly, it focused on generating synonymous samples with the same sentiment label, while our work aims to generate antonymous samples with the reversed sentiment label; Secondly, our discriminator contains an original-side predictor and an antonymous-side predictor which are paired for dual sentiment classification, and alleviate the spurious association problem.

Antonymous sample generation focused on either manually creating antonymous samples (Kaushik et al., 2020; Wang and Culotta, 2021) or resorting to WordNet to generate antonymous samples by replacing some words in the real samples with their antonyms (Xia et al., 2013, 2015a,b). However, these methods primarily rely on human efforts or manually-designed rules, which limits the diversity of generated samples.

Instead of constructing the antonymous samples by human efforts or rules, we aim to propose an end-to-end reinforcement learning framework, for joint counterfactual data generation and dual sentiment classification.

[1] The source code of this work is publicly released at https://github.com/NUSTM/RCDA

3 Approach

Figure 1 illustrates the overall architecture of our framework, which contains two main modules: 1) Antonymous sentence generator. Given an original sentence, the generator replaces each word in the original sentence with one of its antonyms or synonyms from WordNet to generate a number of antonymous sentences as candidates; 2) Dual discriminator. It contains an original-side sentiment predictor and an antonymous-side sentiment predictor, which regards the original and antonymous samples as pairs to perform dual sentiment prediction.

3.1 Antonymous Sentence Generator

The word substitution-based methods have been shown to be effective and stable in synonymous sentence generation. Inspired by Xu et al. (2019), we propose to generate antonymous sentences based on word substitution.

Specifically, we define three word substitution rules for each word in the sentence: no replacement, replacing with an antonym, and replacing with a synonym. Given an input sentence, since its sentiment is often determined by adjectives, adverbs, and verbs, we first utilize WordNet[2] to obtain the antonyms of these three types of words, and replace these words with their antonyms; Second, for nouns and the remaining adjectives, adverbs, and verbs that do not have antonyms, we replace them with their synonyms in WordNet; Lastly, for other words such as stop words, we retain them to avoid irrelevant information. For example, given a sentence "*a good and funny story*", "*good*" and "*funny*" are replaced with their antonyms (e.g., "*bad*" and "*dull*"), and "*story*" is replaced with its synonym (e.g., "*tale*"), and other words are kept. We therefore obtain an antonymous sentence "*a bad and dull tale*".

As WordNet provides multiple synonyms and antonyms for each word, we initialize our generator based on multi-label learning during the warm-up stage.

Formally, given a sequence of input tokens $x = \{w_1, w_2, \cdots, w_n\}$ and its label sequence denoted by $Y = \{y_1, y_2, \cdots, y_n\}$, each token w_t corresponds to a V-dimensional multi-label vector $y_t = \left[y_t^1, \cdots, y_t^j, \cdots, y_t^V\right]$, where V is the size of the vocabulary, and $y_t^j \in \{0, 1\}$ denotes that whether the j-th word in the vocabulary belongs to the set of replacement words for w_t. If the number of replacement word (antonyms or synonyms) in WordNet for w_t is larger than a pre-set threshold K, we select the top-K words with the highest frequency as the supervision signals in multi-label learning.

Specifically, we feed the input sentence to an LSTM encoder, and obtain the hidden representation of each word, denoted by h_t. Next, we feed h_t to V separate binary classifiers:

$$p(y_t^j|w_t) = \text{logistic}\ (W_j h_t + b_j),\ \ j \in 1, \cdots, V. \tag{1}$$

Based on this, we obtain the probability of each vocabulary word belonging to the replacement word set, and re-normalize the probabilities to obtain the multinomial word distribution as follows:

$$P_t = \text{normalize}\ [p(y_t^1 = 1), \cdots, p(y_t^V = 1)]. \tag{2}$$

It should be noted that for vocabulary words that are not inlcuded in WordNet, we set their probabilities in the multinomial distribution to be 0.

Given a training sample (x, s) where s is the sentiment label, we sample a word according to P_t in Eqn. (2) for each word w_t in x as follow:

$$w_t \sim \text{Multinomial}(P_t), \tag{3}$$

and repeat this process to get an antonymous sample: $(\bar{x}, \bar{s})$, where $\bar{s}$ denotes the reversed sentiment label, e.g., positive → negative, or negative → positive. For example, let us assume the distribution of antonyms for "*good*" and "*funny*" are [stale: 0.3, bad: 0.4, displeasing: 0.3] and [serious: 0.2, boring: 0.5, dull: 0.3] respectively, and the synonym distribution of "*story*" is [fiction: 0.2, narration: 0.2, tale: 0.6]. Given a positive sentence "*a good and funny story*", we first sample the antonyms for "*good*" and "*funny*" (e.g., stale and boring), and then sample a synonym for "*story*" (e.g., narration). We therefore obtain an antonymous sentence "*a stale and boring narration*", and set its sentiment label to negative. The process can be repeated to get different antonymous sentences.

According to the method above, a set of antonymous training samples $\{(\bar{x}_i, \bar{s})\}_{i=1}^{M}$ is generated based on an original training sample (x, s).

3.2 Dual Discriminator

Based on the original and the antonymous samples, we construct a dual discriminator, which contains

[2] https://wordnet.princeton.edu/

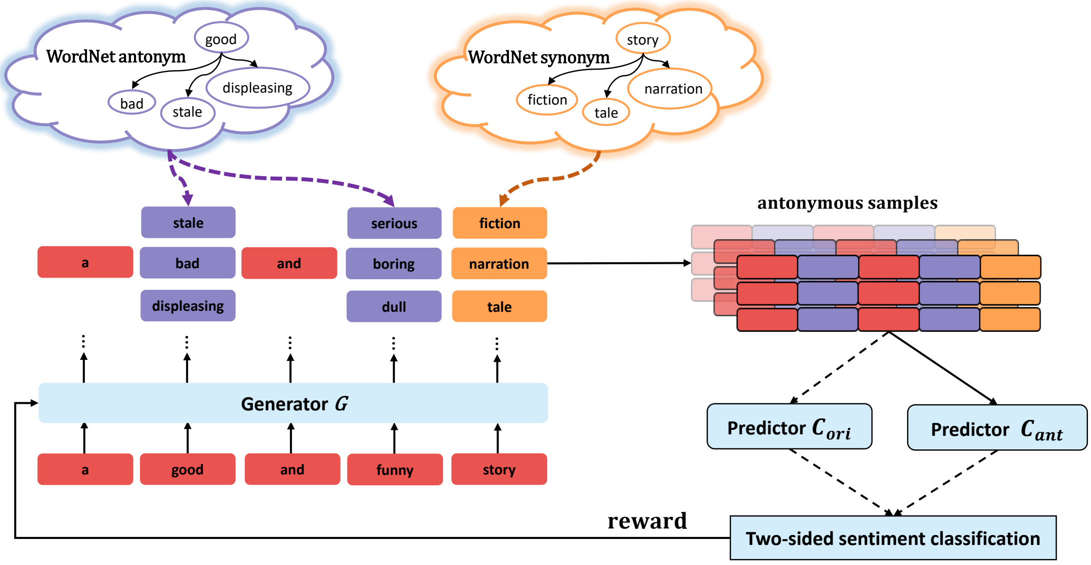

Figure 1: The overall architecture of our joint counterfactual data generation and dual sentiment classification framework. The left part is the generator, which acts as an agent in reinforcement learning, and the right side is the discriminator containing two sentiment predictors, which acts as the environment in reinforcement learning and also serves as the final sentiment classifier at the test stage. The dashed line indicates that there is no back propagation during training.

a pair of predictors: an original-side sentiment predictor C_{ori} and an antonymous-side sentiment predictor C_{ant}. C_{ori} is trained based on the original training set $\mathcal{D}_{ori}$, whose parameters are fixed during reinforcement learning, whereas C_{ant} is trained based on the antonymous training set $\mathcal{D}_{ant}$, whose parameters are incrementally learned and dynamically updated based on the antonymous training set generated in each epoch.

For both C_{ori} and C_{ant}, given the antonymous sentence $\bar{x}$, their hidden representations $\bar{h}_{ori}$ and $\bar{h}_{ant}$ are followed by the softmax layers for dual sentiment predictions respectively:

$$p_{ori}(s|\bar{x}) = \text{softmax}\,(W_{ori}\bar{h}_{ori} + b_{ori}), \quad (4)$$

$$p_{ant}(s|\bar{x}) = \text{softmax}\,(W_{ant}\bar{h}_{ant} + b_{ant}), \quad (5)$$

where W_{ori} and b_{ori} are the parameters for C_{ori}, W_{ant} and b_{ant} are the parameters for C_{ant}. We employ LSTM, BERT-base, and BERT-large (Devlin et al., 2019) as the text encoder in the discriminator.

3.3 Reinforcement Training

To jointly optimize the generator and the discriminator with reinforcement learning, we regard the predictor C_{ori} and the predictor C_{ant} as the environment to get dual sentiment predictions, and to evaluate the quality of the generated samples. We expect that the prediction of $\bar{x}$ from C_{ori} is inconsistent with the original label s, while the prediction from the antonymous sentiment classification module C_{ant} is consistent with $\bar{s}$. For example, given a positive sentence x "*a good and funny story*" and the generated negative one $\bar{x}$ "*a stale and boring narration*", we expect the possibility of $\bar{x}$ being positive to be as small as possible, and the possibility of $\bar{x}$ being negative to be as large as possible. Therefore, we design a new action reward which takes predictions from both C_{ori} and C_{ant} into account:

$$r(\bar{x}) = (1-\alpha)(s - p_{ori}(s|\bar{x})) + \alpha p_{ant}(\bar{s}|\bar{x}), \quad (6)$$

where α is a trade-off parameter. It should be noted that due to the cold start problem of C_{ant}, α is initialized to 0 during the training process of reinforcement learning, and increased to 1 as the performance of C_{ant} increases.

If the reward of $\bar{x}$ is relatively large, our model regards it as a high-quality antonymous sample, and encourages its generation in the next epoch of training, otherwise if the reward is relatively small, our model learns to decrease the possibility of generating it in the next epoch. In policy gradient-based methods, it is a common practice to subtract a baseline reward from the current reward. The goal of the baseline reward r_b is to enforce the generator to select $\bar{x}$ that yields a reward

```
Algorithm 1 RCDA
Require: Generator $G$; Discriminator $C_{ori}$ and $C_{ant}$; dataset $\mathcal{D}_{ori}$
  Randomly initialize the models
  train $C_{ori}$ using $\mathcal{D}_{ori}$
  Warm-up $G$ based on multi-label learning
  for training step do
    Sample $M$ $\bar{x}$ using Eq.(3)
    for $i = 0$ to $M$ do
      Calculate $r^{'}(\bar{x}_i)$ using Eq.(7)
      Compute the loss in Eq.(8)
    end for
    Update the parameters of $G$
    Generate $\bar{x}$ using Eq.(2)
    Use $\bar{x}$ to update the parameters of $C_{ant}$
  end for
  Return the generator $G$; Discriminator $C_{ori}$ and $C_{ant}$
```

$r(\bar{x}) > r_b$ and discourages those that have reward $r(\bar{x}) < r_b$.

In contrast to Xu et al. (2019) that only sampled one synonymous sentence for each sentence and defined r_b as the expectation of the reward of all sampling sentences, we sample M antonymous sentences for each sentence, and use the average value of these M antonymous sentences as the baseline reward $r_b = \frac{1}{M}\sum_{j=1}^{M} r(\bar{x}_j)$. Based on this, we use the following formula to calculate the reward and then feed it to the generator:

$$r^{'}(\bar{x}) = r(\bar{x}) - r_b. \tag{7}$$

Compared with Xu et al. (2019), our reward function ensures that for each original sample, at least one generated antonymous sample is leveraged to optimize model parameters, and these antonymous samples can be regarded as supervisory signals to help the generator generate better antonymous sentences in the next epoch based on the following cost:

$$L = -\log r^{'}(\bar{x}) P_G(\bar{x}|x). \tag{8}$$

Algorithm 1 presents the whole process of our joint counterfactual data generation and dual sentiment classification method.

3.4 Dual Sentiment Classification

In existing antonymous data augmentation approaches, data generation and sentiment classification are often conducted as a pipeline (Kaushik et al., 2020; Wang and Culotta, 2021; Xia et al., 2013, 2015a,b), where a sentiment classification model is separately trained after generating the antonymous samples. In contrast, our reinforcement learning framework integrates antonymous sentence generation and sentiment classification in an end-to-end fashion, and we can also directly use the two sentiment predictors C_{ori} and C_{ant} to perform dual sentiment prediction for testing samples.

Specifically, given an original test sentence x, we first employ the generator G to generate the antonymous test sentence $\bar{x}$, and then use the two predictors C_{ori} and C_{ant} to perform dual sentiment prediction similar as (Xia et al., 2015b):

$$p(s|x) = \begin{cases} p_{ori}(s|x), & \text{if } p_{ori}(s|x) > \min(\tau, p_{ant}(s|\bar{x})) \\ p_{ant}(s|\bar{x}), & \text{otherwise} \end{cases} \tag{9}$$

where $p_{ori}(s|x)$ is the prediction from C_{ori} on x, $p_{ant}(s|\bar{x})$ is the prediction from C_{ant} on $\bar{x}$, and τ is a confidence threshold. In general, the final prediction relies the original predictor when when the confidence of original predictor is higher than that of the antonymous predictor or a threshold ; otherwise the final prediction relies on the antonymous predictor.

It is worth noting that a recent study (Wang and Culotta, 2021) revealed that for antonymous data augmentation approaches, the performance of merging antonymous samples with original samples generally drops when using the antonymous samples generated from rules or machine learning approaches, and it can increase only when using the manually generated samples. In our experiments, we obtain similar observations. The results of using different ways to leverage the antonymous samples are compared in Section 4.4.

4 Experiments

4.1 Experiment Settings

Datasets. We conduct experiments on four benchmark datasets for sentence-level sentiment classification, namely, SST-2, SST-5, RT, and Yelp. SST-2 and SST-5 are the movie reviews from the Stanford sentiment treebank (Socher et al., 2013), which contains both binary and 5-class classification tasks. RT is another sentiment classification dataset containing movie reviews with two labels, released by Pang and Lee (2005). Yelp is a large-scale dataset collected from the Yelp website[3], which contains a large amount of restaurant reviews with rating labels varying from 1 to 5. Following Xu et al. (2019), we sample 100K data as the training set, 10K as the validation set, and 10K for testing.

[3] http://www.yelp.com/dataset/challenge

Settings & Hyperparameters. In the warm up stage, we train the generator for 40 epochs and train the original sentiment predictor for 100 epochs, and then train both the generator and the antonymous sentiment predictor based on reinforcement learning for 60 epochs. For the generator, we set the size of hidden dimension, batch size, learning rate, and sentence sampling times M to 300, 8, 1e-3, and 32, respectively. For the LSTM text encoder, we set the batch size, the size of hidden dimension, the learning rate, the embedding drop rate, and the representation dropout rate to 64, 300, 1e-3, 0.4, and 0.1, respectively. For the BERT text encoder, we set the batch size and the learning rate to 8 and 2e-5. Besides, for τ, we set it as 0.8(0.52) for the two binary classification datasets, and set it as 0.4(0.22) for SST-5 and Yelp when the encoder is LSTM(BERT). All the parameters are optimized with the Adam optimizer, and tuned on the development set of each dataset.

4.2 Compared Systems

We employ LSTM, BERT-base, and BERT-large as our text encoder to systematically evaluate our approach, and compare our Reinforced Counterfactual Data Augmentation (RCDA) approach with the following methods:

- **SynDA** (Zhang et al., 2015), which randomly replaces words in the real samples with synonyms from WordNet to generate synonymous samples.
- **Back-tran** (Sennrich et al., 2016), which translates real to other language via exiting translation model, and then translates it back to source language to get synonymous samples.
- **ConDA** (Kobayashi, 2018), which uses the language model to obtain synonyms for each word and randomly replaces words with these synonyms to obtain adversarial samples.
- **VAT** (Miyato et al., 2017), which improves the model robustness by adding random perturbation to the embedding layer to obtain new adversarial examples.
- **LexicalAT** (Xu et al., 2019), which first uses the generator to randomly replace words with its synonym, hyponym or hypernym to obtain new samples, and then jointly optimizes the generator and the discriminator based on adversarial learning.

Method	SST-2	SST-5	RT	Yelp
LSTM	80.28	39.97	76.03	61.79
+SynDA	80.30	40.20	/	/
+Back-tran	80.77	39.59	76.32	61.76
+ConDA	80.10	40.50	/	/
+VAT	81.16	37.38	75.94	59.69
+DSA	81.32	40.62	75.92	61.23
+AGC	76.00	32.03	71.80	60.53
+LexicalAT	81.60	41.99	76.22	61.18
+RCDA	**82.97**	**42.35**	**78.87**	**62.44**

Method	SST-2	SST-5	RT	Yelp
BERT_{B}	91.52	53.66	87.14	66.17
+Back-tran	91.81	53.93	87.41	65.72
+AGC	89.51	52.76	85.30	65.54
+RCDA	**91.98**	**54.02**	**88.23**	**66.57**

Method	SST-2	SST-5	RT	Yelp
BERT_{L}	92.86	55.25	88.33	66.93
+Back-tran	92.96	54.70	88.21	66.84
+AGC	93.02	53.24	87.69	66.17
+LexicalAT	93.03	55.38	88.68	**67.50**
+RCDA	**93.30**	**55.62**	**89.07**	67.41

Table 1: The accuracy of compared systems on four benchmark datasets for sentence-level sentiment classification, where BERT_{B} and BERT_{L} refer to BERT-base and BERT-large respectively.

- **DSA** (Xia et al., 2015b), which first replaces original words with their antonyms from WordNet, and then employs the original and antonymous samples for dual sentiment analysis under softmax regression.
- **AGC** (Wang and Culotta, 2021), which first uses WordNet to obtain antonyms for N most important words in the corpus, and then uses the word substitution method to obtain counterfactual samples to improve the model robustness.

4.3 Main Results

The results of our proposed approach and compared systems are shown in Table 1. We can easily observe that our RCDA method consistently outperforms all the compared systems by using LSTM, BERT-base, and BERT-large as our text encoder.

Specifically, for the LSTM text encoder, RCDA outperforms the baseline approach by around 2 absolute percentage points on accuracy for each data set. For the BERT text encoder, RCDA outperforms BERT-base by 0.46% on SST-2, 0.36% on SST-5, 1.09% on RT, 0.4% on Yelp, respectively. Although BERT-large already reaches highly competitive results, our RCDA approach can still significantly boost its performance across the four datasets.

Moreover, we can easily observe that our RCDA approach consistently outperforms most existing data augmentation-based methods including SynDA, ConDA, VAT, DSA, and AGC across the four datasets. In addition, even in comparison with one of the state-of-the-art data augmentation approach LexicalAT, our RCDA method can generally achieve better performance across four datasets, except when using BERT-large as the text encoder. We confirm that the improvements are significant according to the paired t-test.

All these observations demonstrate the effectiveness and robustness of our proposed RCDA approach.

4.4 In-depth Analysis

The effect of alleviating spurious association. In order to evaluate whether our generated antonymous samples can alleviate the spurious association problem, we use word frequency as features to train a logistic regression model for the SST-2 dataset, and observe the coefficient changes of neutral words before and after adding antonymous samples to the training data. Take "*English*" as an example, because it has a higher word frequency in positive class than the negative class, its coefficient in the original classifier is a positive value (0.5838). After incorporating antonymous samples, its coefficient drops from 0.5838 to 0.1231. Similar trends have been observed for other neutral words such as "*book*", "*movie*", "Chinese" and so on, as shown in Table 2. It demonstrates that the incorporation of antonymous samples can alleviate the spurious association between neutral words and the class labels.

Word	Original Coefficient	New Coefficient
book	-0.3719	-0.1477
English	0.5838	0.1231
Chinese	0.5791	-0.0927
movie	-0.2460	-0.0175

Table 2: The coefficients of words before and after generating antonymous samples.

Diversity of the generated antonymous samples. We further evaluate the diversity of antonymous samples generated by different approaches under the evaluation metric named *distinct-2* (Li et al., 2016). In Table 3, it can be observed that the diversity of antonymous samples generated by our RCDA approach is significantly larger than AGC and DSA, because AGC and DSA used fixed rules for antonymous sentence generation. This indicates that our method can indeed generate more diverse antonymous samples than previous approaches. Moreover, for each original sentence, our RCDA approach can automatically generate multiple antonymous sentences, instead of generating only one antonymous sentence.

	SST-2	SST-5	RT	Yelp
DSA	0.543	0.520	0.524	0.134
AGC	0.561	0.543	0.542	0.138
RCDA	**0.567**	**0.555**	**0.554**	**0.143**

Table 3: Comparisons on the diversity of antonymous samples generated by different approaches.

The effect of reinforcement learning for antonymous sentence generation. To demonstrate the effectiveness of reinforcement learning for antonymous sentence generation, we consider a simple compared system named Random, i.e., randomly selecting candidate words to build antonymous samples, followed by using our dual sentiment predictors to make the final sentiment classification. Moreover, we also report the accuracy of only using the antonymous sentiment predictor for prediction. As shown in Table 4, our RCDA method consistently outperforms the random sampling approach for both dual sentiment predictor and the antonymous sentiment predictor. This implies that reinforcement learning can gradually filter out the low-quality antonymous samples, and select the best antonymous samples for dual sentiment classification.

LSTM	**SST-2**	**SST-5**	**RT**	**Yelp**
Random-ant	78.58	38.69	74.46	60.92
RCDA-ant	80.72	40.32	76.95	61.59
Random	81.76	40.99	76.67	62.08
RCDA	**82.97**	**42.35**	**78.87**	**62.44**
BERT$_{\text{B}}$	**SST-2**	**SST-5**	**RT**	**Yelp**
Random-ant	80.14	40.20	77.72	62.54
RCDA-ant	81.32	41.54	79.90	63.16
Random	91.65	53.59	87.25	66.40
RCDA	**91.98**	**54.02**	**88.23**	**66.57**

Table 4: The impact of reinforcement learning. "ant" refers to only using the antonymous sentiment predictor for prediction.

Sensitivity analysis of M. In order to analyze the impact of the number of sampling samples M in Section 3.3, we further conduct experiments on the antonymous samples by varying the values of M for SST-2, SST-5, and RT datasets, respectively.

Problem	Text	Confidence	Prediction	Daul prediction
Out of vocabulary word	**Original:** escapism in its purest form . **Antonymous:** escapism in its impure work .	(0.9513, **0.0487**) (**0.9978**, 0.0022)	negative ✗ negative ✓	positive ✓
Low frequency word	**Original:** a trashy , exploitative , thoroughly unpleasant experience . **Antonymous:** a valuable , generative , thoroughly pleasant inexperience .	(**0.2619**, 0.7381) (0.0002, **0.9998**)	positive ✗ positive ✓	negative ✓
Ambiguous sentiment word	**Original:** all but the most persnickety preteens should enjoy this nonthreatening but thrilling adventure . **Antonymous:** some but the fewest humble preteens should suffer this nonthreatening but unexciting venture .	(0.8189, **0.1811**) (**0.9997**, 0.0003)	negative ✗ negative ✓	positive ✓

Table 5: Several antonymous samples generated by our method. The bold confidence dimension is the correct label. With the help of the antonymous samples, our dual sentiment classification method made correct predictions.

Experimental results in Figure 2 show that the initial increase of M gradually improves the performance of the antonymous sentiment classifier; the best performance can be generally observed when M=32; after that, the performance gradually drops as M increases. Therefore, we set M as 32 in our main experiments.

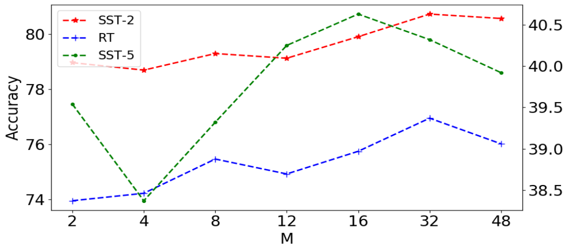

Figure 2: The impact of different values of M on the antonymous sentiment predictor.

Sensitivity analysis of K. We further analyze the impact of the value of the maximum number of antonym (or synonym) substitution (i.e., K in Section 3.1) on the SST-2, SST-5 and RT datasets. Figure 3 shows that the model can achieve the best performance when K is around 3. Specifically, when K is relatively small, the diversity of the sample is poor; when K is relatively large, words with small or even zero word frequency may be introduced into the generated antonymous samples, which will drop the performance of the sentiment classifier. Based on the result, we set K to 3 across all the datasets.

4.5 Case Study

Finally, to better understand the advantage of the generated antonymous samples, we display several representative test samples in Table 5, for which the original sentiment predictor made wrong predictions, while the antonymous sentiment predictor made correct predictions. These samples can be grouped into three categories, i.e., containing out of vocabulary words, low frequency words, and ambiguous sentiment words.

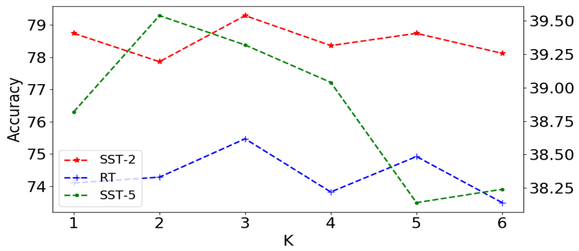

Figure 3: The impact of different values of K on the antonymous sentiment predictor.

Based on the first example, it can be found that antonymous samples can solve the *out of vocabulary* issue. In the original sample, since "*purest*" is an out of vocabulary word, the prediction from the original predictor is wrong. But in the antonymous sample, "*purest*" is replaced with "*impure*" which occurred many times in the training set. Therefore, the antonymous predictor made the correct prediction.

In the second example, although the original sample contains three negative sentiment words, their word frequency is relatively low in the training set, which leads to the incorrect prediction of the original predictor. In contrast, in the antonymous sample, these rare words are replaced with frequent antonymous words such as "*valuable*" and "*pleasant*", which helps correct the incorrect prediction.

Finally, for the third example, as "*thrilling*" is a word with ambiguous sentiments, the original predictor gave incorrect predictions. In the antonymous sample, "*thrilling*" is replaced by a negative word "*unexciting*", which helps our model correctly predict its sentiment.

5 Conclusion

In this paper, we propose an end-to-end reinforcement learning framework named Reinforced Counterfactual Data Augmentation (RCDA) for joint counterfactual data augmentation and dual sentiment classification, to address the over-fitting problem and improve the generalization ability of sentiment classification models. RCDA contains an antonymous sentence generator to automatically generate massive diverse antonymous sentences and a dual discriminator with an original-side sentiment predictor and an antonymous-side sentiment predictor, which are jointly optimized based on our reinforcement learning framework. Experiments on four benchmark datasets show that our approach consistently outperforms strong data augmentation baselines. In-depth analysis demonstrates the advantage of our approach in generating diverse training data and alleviating the spurious association problem.

Acknowledgments

This work was supported by the Natural Science Foundation of China (No. 62076133 and 62006117), and the Natural Science Foundation of Jiangsu Province for Young Scholars (No. BK20200463) and Distinguished Young Scholars (No. BK20200018).

References

Danilo Croce, Giuseppe Castellucci, and Roberto Basili. 2020. GAN-BERT: generative adversarial learning for robust text classification with a bunch of labeled examples. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2114–2119.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics (ACL)*, pages 4171–4186.

Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1875–1885.

Rie Johnson and Tong Zhang. 2017. Deep pyramid convolutional neural networks for text categorization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 562–570.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomás Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 427–431.

Divyansh Kaushik, Eduard H. Hovy, and Zachary Chase Lipton. 2020. Learning the difference that makes A difference with counterfactually-augmented data. In *8th International Conference on Learning Representations (ICLR)*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Sosuke Kobayashi. 2018. Contextual augmentation: Data augmentation by words with paradigmatic relations. In *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 452–457.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 110–119.

Takeru Miyato, Andrew M. Dai, and Ian J. Goodfellow. 2017. Adversarial training methods for semi-supervised text classification. In *5th International Conference on Learning Representations (ICLR)*.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 115–124.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th Conference of the Association for Computational Linguistics (ACL)*, pages 1085–1097.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 86–96.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642.

Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1556–1566.

Zhao Wang and Aron Culotta. 2021. Robustness to spurious correlations in text classification via automatically generated counterfactuals. In *Proceedings of the 35th Conference on Artificial Intelligence (AAAI)*, pages 14024–14031.

Rui Xia, Cheng Wang, Xin-Yu Dai, and Tao Li. 2015a. Co-training for semi-supervised sentiment classification based on dual-view bags-of-words representation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1054–1063.

Rui Xia, Tao Wang, Xuelei Hu, Shoushan Li, and Chengqing Zong. 2013. Dual training and dual prediction for polarity classification. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 521–525.

Rui Xia, Feng Xu, Chengqing Zong, Qianmu Li, Yong Qi, and Tao Li. 2015b. Dual sentiment analysis: Considering two sides of one review. *IEEE transactions on knowledge and data engineering*, pages 2120–2133.

Xiaoyu Xing, Zhijing Jin, Di Jin, Bingning Wang, Qi Zhang, and Xuanjing Huang. 2020. Tasty burgers, soggy fries: Probing aspect robustness in aspect-based sentiment analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3594–3605.

Jingjing Xu, Liang Zhao, Hanqi Yan, Qi Zeng, Yun Liang, and Xu Sun. 2019. Lexicalat: Lexical-based adversarial reinforcement training for robust sentiment classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5521–5530.

Ruiyi Zhang, Changyou Chen, Zhe Gan, Wenlin Wang, Dinghan Shen, Guoyin Wang, Zheng Wen, and Lawrence Carin. 2020. Improving adversarial text generation by modeling the distant future. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2516–2531.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems*, pages 649–657.