Data Augmentation with Sentence Recombination Method for Semi-supervised Text Classification

Anonymous ACL submission

Abstract

As the need of large amount of time and expertise to obtain enough labeled data, semi-supervised learning has received much attention to utilize both labeled and unlabeled data. In this paper, we present **SeRe**: a **Se**ntence **Re**combination method to augment training data for semi-supervised text classification. SeRe makes full use of the similarities between sentences in different samples through the grouping and recombining process to form rich and varied training data. SeRe generates data from three combinations, including labeled, unlabeled, and mixed data. Meanwhile, SeRe combines the self-training framework to improve the quality of augmented training data iteratively. We apply SeRe to text classification tasks and conduct extensive experiments on four publicly available benchmarks. Experimental results show that SeRe achieves new state-of-the-art performances on all of them.

1 Introduction

In recent years, deep learning methods have achieved good results in natural language processing (Devlin et al., 2018; Yang et al., 2019b; Li et al., 2020; Zhong et al., 2020; Zhang et al., 2019). However, deep learning methods often rely on the supervision information of the training set. The collection of the training samples often requires high manual labelling costs. Due to the difficulty of data acquisition or labeling, people can only obtain small-scale labeled data in many cases. Under the limited amount of labeled data, the neural network is prone to overfitting and poor generalization. Compared with labeled data, unlabeled data is easier to obtain and collect. A series of researchers have devoted themselves to the research of semi-supervised learning tasks (Lee et al., 2013; Miyato et al., 2018; Xie et al., 2019; Chen et al., 2020). They use a small amount of labeled data and a large amount of unlabeled data to train models. Under the premise of saving many labeling costs, performance close to the fully-supervised model is achieved.

In the task of text classification, a family of semi-supervised works (Xie et al., 2019; Chen et al., 2020; Liu et al., 2021) is proposed. Most of these methods focus on constructing a loss function to train labeled and unlabeled data jointly. Although these methods have shown to be effective, we argue that they do not fully use the supervisory information from labeled data and the diversity of the features provided by unlabeled data.

Data augmentation technology is a useful solution to solve the shortage of training data. Existing methods (Cubuk et al., 2018; Lemley et al., 2017; Perez and Wang, 2017) have achieved excellent results on visual tasks. In the natural language processing task of text classification, however, augmentation is more difficult than in visual tasks due to the discrete attributes between words and sentences. The method (Sennrich et al., 2015) is proposed based on back-translation to change the expression form of each sentence. (Wei and Zou, 2019) proposed to slightly disturb the text based on addition, deletion, and modification. Although these methods expand the amount of data to a certain extent and improve the model's performance, back translation and perturbation operations may affect the sentence information and even destroy the grammatical structure. Moreover, the sample richness of the dataset formed by existing augmentation methods is insufficient.

In this paper, addressing the challenges in semi-supervised text classification tasks and the shortcomings of existing methods, we propose a novel data augmentation method called SeRe. We expand the sample size and diversity of the dataset by regularly reorganizing the sentences in the dataset to form new augmented samples. The training set for the semi-supervised text classification task contains both labeled and unlabeled data. For the labeled data, we group the samples according to the labels.

Furthermore, for the samples in each label, such as label c, we use the pre-trained model to inference the sentences in the samples. According to the output confidence of class c, we distribute them into True Bucket and False Bucket. Then, we select sentences with similar semantics in the same bucket at random and swap positions, resulting in disturbed samples as augmented data. The buckets organize these sentences into different classes in order to smooth the disturbance amplitude as much as possible so that the augmented data does not have a negative impact on model training. For the unlabeled data, We first generate pseudo-labels using the model that was previously trained on labeled data, and then we filter out high-quality samples based on the confidence ranking. To augment unlabeled data, we use the same method as we do with labeled data. In order to make full use of labeled and unlabeled data, we further replace the labeled sentences with the unlabeled ones according to the rules of the proposed augmentation method to form the enriched augmented data, which serves as the mixed data. Finally, we introduce the augmentation procedure into a self-training framework which iteratively conducts augmenting, selecting, and training steps.

The main contributions of this work can be summarized as follows: 1) We propose a novel data augmentation method through sentence recombination for semi-supervised text classification; 2) To fully leverage the labeled and unlabeled data, we augment the training data from three combinations (labeled, unlabeled, and mixed) with a self-training framework; 3) We conduct extensive experiments on four widely-used text classification benchmarks: IMDb (Maas et al., 2011), AG-News (Zhang et al., 2015), Yahoo! Answers (Chang et al., 2008), and Yelp-5 (Zhang et al., 2015). Experimental results show the effectiveness of the proposed method.

2 Related Work

2.1 Data Augmentations for Text

Because data collection and labeling require much time, the data used to train the model in many scenarios is very limited. Under this limitation, it has become a powerful solution to expand based on existing data by expanding the amount of data and increasing the diversity of data. In the field of computer vision, some works (Cubuk et al., 2018; Lemley et al., 2017; Perez and Wang, 2017) is devoted to the use of images through operations such as shifting, zooming in/out, rotating, flipping to generate disturbing data to improve data diversity. There is also work to improve the quality of training by combining different images to form new ones. However, in natural language processing, the augmentation of text data has become a challenging research field due to the discrete nature of text data and its unique semantic structure.

(Wei and Zou, 2019) proposed some local disturbance strategies for augmentation. Various operations, including synonym replacement, random insertion, random swap, and random deletion, are used to modify text data. However, this approach essentially destroys the sentence structure and even produces grammatical errors, making it difficult to control the disturbance amplitude and reducing performance due to negative examples. On the basis, (Karimi et al., 2021) proposed a more straightforward augmentation method by randomly inserting punctuation into sentences. However, this approach is trivial and does not form truly diverse data. Different from this kind of method, (Sennrich et al., 2015) proposed back-translation to generate new expressions of text data. This type of method expands text data diversity by translating sentences into other languages and then recovering them. However, this type of method is highly dependent on the translation quality, and it is also easy to cause the data to fall into a situation where the semantics are destroyed. In the work (Chen et al., 2020), a type of soft-label-based method was proposed to combine the representation features of sentences in the hidden layer of the model and indirectly expand the data diversity. This type of method was first proposed in the field of computer vision. Since the original data has not been modified, this method does not improve the quality and diversity of the data.

2.2 Semi-Supervised Learning on Text

Due to the difficulty of collecting and labeling data in some scenarios, semi-supervised learning has received widespread attention in the field of deep learning (Lee et al., 2013; Miyato et al., 2018; Xie et al., 2019; Chen et al., 2020). Compared with labeled data, unlabeled data is easier to obtain and highly diverse. (Lee et al., 2013) proposed constructing pseudo-labels on unlabeled data for supervised training. (Yang et al., 2017) used autoencoder (VAE) to model from sequence to sequence and made progress in semi-supervised text clas-

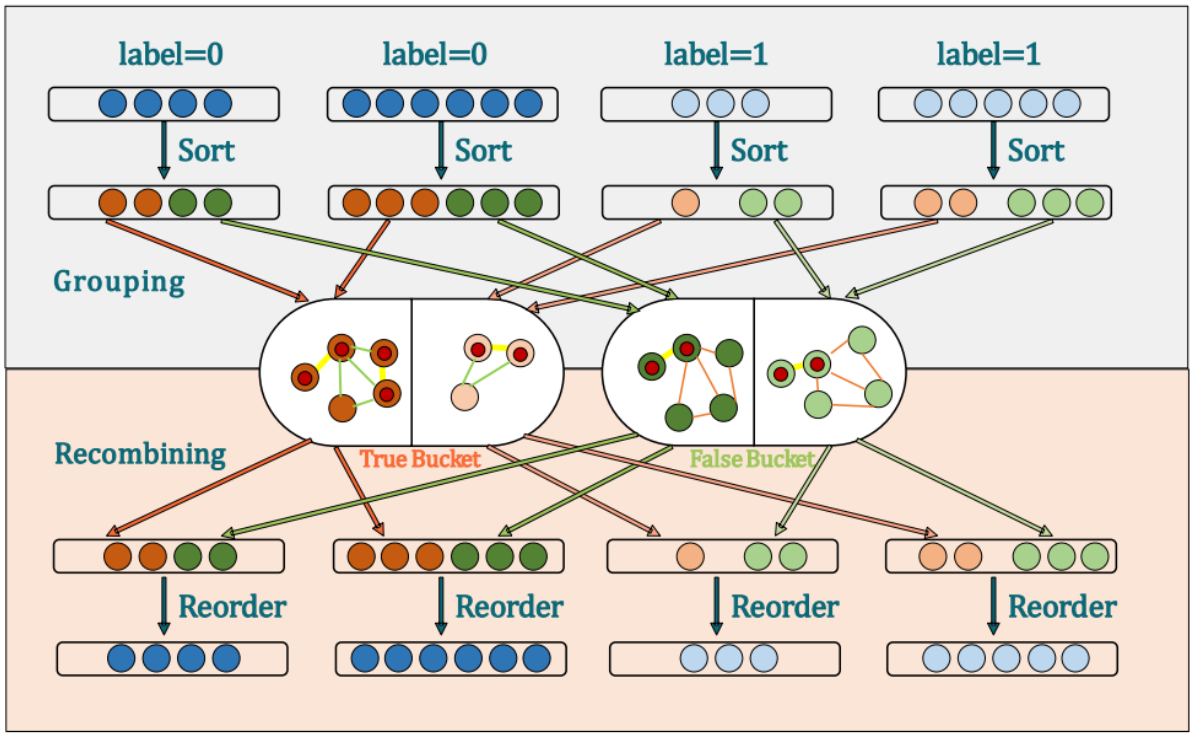

Figure 1: An illustration of the proposed SeRe. The upper part shows the Grouping process, and the lower part shows the Recombining process. Each bar in the figure represents a sample, and the circles inside represent sentences. The dark circles represent sentences with label 0, and the light ones represent sentences with label 1. The red circles represent the sentences with high confidence in a sample (assigned to True Bucket), and the green ones represent the sentences with low confidence (assigned to False Bucket).

sification tasks. (Miyato et al., 2017) introduced virtual adversarial training technology into the field of natural language processing. This method forms the data disturbance by embedding words into sentences. (Yang et al., 2019a) proposed a hierarchical processing method according to the quality of data labels. They hand over low-quality labels to high-quality label supervision and train the model hierarchically. (Berthelot et al., 2019) constructed the consistency loss by perturbing the unlabeled data multiple times and using the model to predict the average value of the data. (Kurakin et al., 2020) analyzed the difference between the predicted distribution and the ground truth distribution and construct loss functions. (Chen et al., 2020) proposed the TMix data augmentation method and combined it with the method of assigning weight to unlabeled data to construct loss functions for semi-supervised text classification tasks. (Liu et al., 2021) introduced an inspirer network together with the consistency regularization framework, which leveraged a generalized regular constraint on the lightweight models for efficient semi-supervised learning. Most of these methods are innovative in model structure and loss function design, but they do not fully use the supervision information of labeled data and the diversity of the features of unlabeled data.

3 Sentences Recombination Approach

In order to improve the diversity of training data, a data augmentation method based on Sentence Recombination (SeRe) is proposed to make full use of the combination relationship between different sentences. The proposed method aims to exchange sentences with similar meanings in different samples to form new combinations. Intuitively, we classify and group all the sentences in the training samples and then recombine them into new samples. Although the augmented data is still based on the sentences in the original training data, the augmented samples formed by partial recombination can be regarded as a kind of perturbation form of the original training data. In the process of recombination, we exchange the sentences with the closest representations in the hidden layer. In this way, the disturbance amplitude can be better controlled, and the negative effect brought by the augmentation

can be effectively suppressed.

3.1 True-False Buckets

The proposed approach regards the samples of the training data as a set of sentences as unit elements. Through the strategic recombination of the elements in the set, new samples are formed. Randomly combining sentences is a relatively straightforward strategy, but it will cause semantic discontinuities between sentences and be inconsistent with the labels. In order to better combine sentences, we propose **True-False Buckets** data structure to group all sentences first. The True Bucket stores the sentences that are classified as corresponding labels with higher confidence in each sample, and the False Bucket stores the sentences that are classified as corresponding labels with lower confidence.

B_{true}, B_{false} are used to represent the set of True Bucket and False Bucket respectively. They each contain C subsets, where C represents the total number of classes of a text classification task. Mathematically, $B_{true} = \{B^1_{true}, ..., B^C_{true}\}, B_{false} = \{B^1_{false}, ..., B^C_{false}\}$. For a subset B^i_{true}, the elements (sentences) contained in it are all from the training samples with the label i. That is $B^c_{true} = \{s_1, s_2, ..., s_{|B^c_{true}|}\}, c \in [1, C]$. On the other hand, the same is true for the elements in the False Bucket.

3.2 Grouping

For a training set $D = (x_i, y_i), i \in (1, ..., n)$, we divide it into multiple subsets according to labels $D = \{D_1, ..., D_C\}$, where C represents the number of classes in a text classification task. Specifically, each subset $D_c = \{t^c_1, ..., t^c_{|D_c|}\}$ contains all the samples labeled with c in the training set, where $c \in (1, ..., C)$. SeRe aims to classify the sentences labeled with a particular class (take c as an example) into two categories. One contains sentences that contribute more to the classification process in the samples, and the other contains the sentences that contribute less to the classification process. The contribution of a sentence is defined as the classification confidence for label c, which is obtained by the forward propagation process with the pretrained model.

For a text $t^c = \{s_1, ..., s_m\}$ consisting of m sentences, we group them into True Bucket and False Bucket respectively according to the following rules. In the following formulas, $g(\cdot)$ represents the inference network pretrained on the original dataset, which outputs a one-dimensional vector, representing the confidence of each class.

$$g(s_i)[c] \geq Median^m_{i=1} g(s_i)[c] \quad (1)$$

For sentences satisfying (1), we add them to True Bucket. Specifically, add them to B^c_{true}, as sentences in Buckets are stored separately by labels. That is to say, the higher the output confidence that a sentence s_i is classified as class c, we roughly think that the sentence contributes more to the entire classification result.

$$g(s_i)[c] < Median^m_{i=1} g(s_i)[c] \quad (2)$$

Similarly, for sentences satisfying (2), we add them to the False Bucket B^c_{false}. We roughly think that these sentences have a relatively small impact on classifying the entire sample into c. Keeping the size of the two buckets equal can effectively avoid data asymmetry and deviation caused by other threshold-based strategies. The grouping scheme is to improve the quality of the following sentence recombining process. See Fig. 1 for more details.

3.3 Recombining

In this subsection, we introduce the approach for recombining sentences and generating augmented samples. We group according to the label and confidence of each sentence in order to control the extent of augmented data modification and prevent the augmented sample from having a large impact on prediction. Therefore, the proposed disturbance effect occurs in each subset of B_{true} and B_{false}.

Taking True Bucket B_{true} as an example, all sentences in each subset B^c_{true} are considered to play a similar role in the classification task, where $c \in (1, ..., C)$. That is to say, the sentences in B^c_{true} have a positive effect on the samples labeled as c. We randomly find several pairs of sentences with the closest semantics in the B^c_{true} set. Then we exchange their positions in the original text, and the semantic similarity is measured by the Euclidean distance between the representation vectors of the last hidden layer. Specifically, we randomly select a sentence s_i in the set of B^c_{true} iteratively and find the sentence s_j with the smallest distance from its representation vector in the set as its replacement object. In order to improve time efficiency, we use the KD-tree structure to reduce the complexity of a match from $O(n)$ to $O(k \log_2 n)$, where k represents the dimension of the vector. See Fig. 1 for more details.

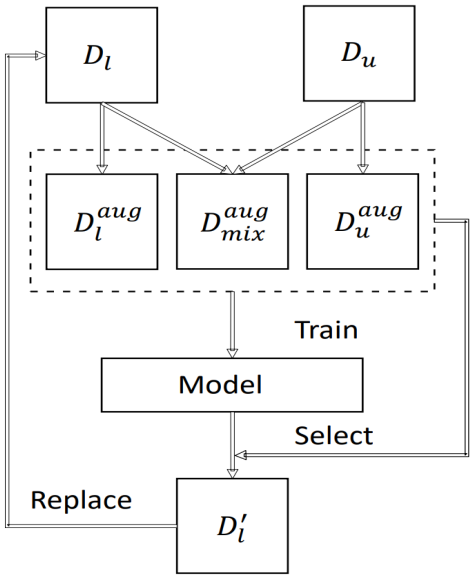

Figure 2: An illustration of the proposed self-training process for semi-supervised tasks. D_l^{aug}, D_u^{aug} represent the data augmented by D_l, D_u respectively, and D_{mix}^{aug} is obtained by mixing D_l and D_u. The labeled dataset D_l in each round is replaced with the subset D_l' selected by the model in the previous round.

4 Semi-supervised Framework

This section shows the important role of the proposed augmentation method in semi-supervised text classification tasks. The training data is divided into labeled and unlabeled in such settings. The labeled dataset $D_l = \{(x_1^l, y_1^l), ..., (x_n^l, y_n^l)\}$ tends to contain a small amount of data, while the unlabeled dataset $D_u = \{x_1^u, ..., x_m^u\}$ generally contains more samples, i.e. $m > n$. Our goal is to use the label information in D_l and the feature diversity in D_u to train text classification models with better effects through data augmentation and selection approaches. To this end, we adopt a self-training strategy to execute the augmentation and section processes in turn iteratively. The framework allows SeRe to increase data quality while enhancing data diversity. On the one hand, the data augmentation method effectively improves the diversity of data and fully integrates the features of labeled and unlabeled data. On the other hand, the data selection process improves the quality of training data, improving the performance of the models. The overall flowchart is shown in Fig. 2.

4.1 Data Augmentation

To fully use the label information in D_l and the feature diversity in D_u, we generate three types of augmented data for training. One is **labeled data augmentation**, which aims to augment the samples in D_l. The second is **unlabeled data augmentation**, which aims to pseudo-label and augment unlabeled samples. The third is to use sentences in D_u to perturb the samples in D_l to form new samples, called **mixed data augmentation**.

4.1.1 Labeled Data Augmentation

Limited labeled data can cause over-fitting problems in model training. Therefore, we use the proposed sentence recombination method SeRe to augment the samples in D_l. In each process, we obtain augmented data of the same scale as the original dataset. This process is repeated to get enough copies as the same amount of unlabeled samples. Due to the randomness in the augmentation process, the data of each round of augmentation has a strong diversity. As shown in Fig. 2, where D_l^{aug} is the dataset augmented by D_l.

4.1.2 Unlabeled Data Augmentation

Unlabeled data contains rich and diverse semantic features, thus we use SeRe to augment D_u. Before augmentation, the unlabeled data is pseudo-labeled by the pre-trained model. Since the model pre-trained on the labeled data has classification ability, the quality of the labeled pseudo-label is guaranteed to a certain extent. In addition, we select a half-size subset of D_u with the highest prediction confidence for augmentation. The method is the same as the labeled data augmentation in the previous subsection. As shown in Fig. 2, where D_u^{aug} is the dataset augmented by D_u.

4.1.3 Mixup Augmentation

We propose an approach for mixing and augmenting labeled data with unlabeled data, called Mixup Augmentation. This approach follows the sentence recombination augmentation method proposed in the previous subsection, replacing sentences in labeled samples with unlabeled sentences. The unlabeled samples are grouped by pseudo-labels and mixed with the buckets of labeled data to form mixed Buckets. Random disturbance occurs in the mixed Buckets to form mixed augmented data. The motivation of this approach is to merge the use of the supervised information of labeled data and the sentence feature diversity of unlabeled data to form high-quality augmented data. As shown in Fig. 2, D_{mix}^{aug} is the dataset after mixing and augmenting D_l, D_u.

4.2 Self-Training

Self-training has demonstrated outstanding performance in a series of natural language processing tasks (Du et al., 2021; Meng et al., 2021; Wang et al., 2021). In order to better play the role of SeRe, we refer to the self-training framework. As shown in Fig. 2, according to the method proposed in the three subsections above, data augmentation is performed on D_l, D_u to obtain $D_l^{aug}, D_u^{aug}, D_{mix}^{aug}$. The three types of augmented data are combined to obtain the training data of the current round. The trained model will be further used to filter the training data. The model's confidence of the corresponding class is used as the basis for selection. Samples with high confidence are thought to be of high quality. The classification confidence of the samples is sorted from high to low, and the highest $|D_l|$ samples are used as the new round of labeled data D_l' ($|D_l|$ represents the number of samples in D_l). The overall self-training flowchart is shown in Fig. 2.

Although data augmentation algorithms can increase the diversity and scale of data to some extent, there will always be low-quality or even harmful data. The model and the data form a closed loop that complements each other by iteratively "augment-train-select" in the self-training training mode. In other words, high-quality data drives the training of high-performance models. Furthermore, high-performance models have more robust selecting capabilities, which further improve the quality of the data.

5 Experiments

In this section, we compare the performance with recent data augmentation and semi-supervised text classification methods. The experiments are conducted on four datasets. The text content includes multiple topics. In the following subsections, we expand from the experimental datasets, implementation details, and quantitative results.

5.1 Datasets

We evaluate the performance of the proposed method on four public datasets IMDb (Maas et al., 2011), AG-News (Zhang et al., 2015), Yahoo! Answers (Chang et al., 2008), and Yelp-5 (Zhang et al., 2015). We split different amounts of data from the original dataset as labeled training data and use the full original test set to evaluate the performance of the methods. For semi-supervised experiments, we follow the data pre-processing proposed in (Chen et al., 2020). The statistics of datasets are shown in Table. 1.

Dataset	Classes	Unlabeled	Dev
IMDb	2	3000	2000
AG-News	4	3000	2000
Yelp-5	5	3000	1000
Yahoo	10	3000	500

Table 1: Statistics and split of IMDb, AG-News, Yahoo! Answers and Yelp-5 for semi-supervised experiments. The number in this table means the number of data per class.

5.2 Implementation Details

For SeRe, a BERT-based-uncased tokenizer is used in this work to tokenize the text. We used the pre-trained bert-based-uncased model and finetuned it for the classification tasks. A two-layer MLP with 768 hidden states and tanh as the activation function is used to predict the labels. In all the experiments, we use Adam to optimize the parameters of each model, and 2e-5 as the learning rate for the BERT encoder, and 1e-3 for the MLP model. In the KD-tree part of the augmentation algorithm, the representation vector used to calculate the distance is defined as the average pooling result of the output by the MLP layers.

5.3 Results

We first construct experiments to verify the effectiveness of the proposed SeRe for semi-supervised tasks. We compared a series of methods on semi-supervised tasks in recent years, including (BERT (Devlin et al., 2018), UDA (Xie et al., 2019), MixText(TMix) (Chen et al., 2020), and FLiText (Liu et al., 2021)). Then, we compared the performance of a series of data augmentation algorithms EDA (Wei and Zou, 2019), AEDA (Karimi et al., 2021), TMix (Chen et al., 2020) on four datasets with different numbers of labeled samples. In this experiment, all the methods only use labeled data.

Comparison with semi-supervised baselines. Table. 2 shows the performance of SeRe and semi-supervised baselines on four benchmarks. We use three types of state-of-the-art models (UDA (Xie et al., 2019), MixText(TMix) (Chen et al., 2020) and FLiText (Liu et al., 2021)) as baselines to test classification performance on four different

Model	IMDb		AG-News		Yelp-5		Yahoo!	
	200	1000	200	1000	200	1000	200	1000
BERT	87.14	89.92	87.65	90.21	56.93	59.70	69.75	72.58
TMix	88.33	90.56	87.95	90.87	57.20	60.02	70.19	72.88
UDA	89.12	90.88	88.20	91.44	59.40	60.92	70.47	73.72
MixText	89.52	91.62	89.68	92.04	58.21	60.44	71.55	73.92
FLiText	89.66	91.20	88.92	91.40	59.48	60.28	70.88	72.29
SeRe	**90.75**	**92.95**	**90.49**	**92.56**	**59.90**	**60.90**	**72.39**	**74.19**

Table 2: The experimental results on the four datasets are expanded in the table, with IMDb, AG-News, Yelp-5, and Yahoo! listed from left to right. For each dataset, two sets of experiments are run; the numbers 200 and 1000 represent the number of labeled samples (per class). Each row in the table shows the performance of a set of baselines. SeRe is the method proposed in this paper. It should be pointed out that TMix only performs data augmentation and training on labeled data and does not use unlabeled data. All other methods must use both labeled and unlabeled data for training.

Dataset	Model	200	1000	Dataset	Model	200	1000
IMDb	BERT	87.14	89.92	Yelp-5	BERT	56.93	59.70
	+EDA	88.24	90.20		+EDA	57.31	60.15
	+AEDA	88.40	90.07		+AEDA	57.95	60.48
	+TMix	88.33	90.56		+TMix	57.20	60.02
	+SeRe	**89.35**	**91.70**		+SeRe	**58.75**	**61.04**
Yahoo!	BERT	69.75	72.58	AG News	BERT	87.65	90.21
	+EDA	69.90	73.06		+EDA	87.90	90.66
	+AEDA	70.52	73.15		+AEDA	88.09	90.95
	+TMix	70.19	72.88		+TMix	87.95	90.87
	+SeRe	**71.24**	**73.85**		+SeRe	**88.59**	**91.78**

Table 3: Performances (%) across four text classification tasks for models with different data augmentation methods on different training set sizes.

datasets. Each dataset is randomly selected 200, 1000 samples per class as labeled data, and 3000 samples per class for unlabeled data. The verification set and test set are as defined in the previous section. All methods in the experiment, except SeRe, are based on model architecture and loss function design. SeRe is committed to improving model performance by improving data quality and diversity. The results show that SeRe achieves state-of-the-art performance on all benchmarks.

Comparison with data augmentation baselines.

Table. 3 shows the performance of SeRe and data augmentation baselines. We use different augmentation methods (EDA (Wei and Zou, 2019), AEDA (Karimi et al., 2021), and TMix (Chen et al., 2020)) to conduct experiments on four datasets. Each dataset is randomly selected 200, 1000 samples per class as the training data and the verification set, and the test set are as defined in the previous section. BERT (Devlin et al., 2018) was selected as the basic model to show performance results without augmented data. For augmented data, each training sample is augmented four times. That is, the size of the training data becomes five times the original. As shown in Table. 3, SeRe has shown superior performance on different datasets. The performance of EDA and AEDA on small datasets (200 per class) is better than that on large datasets (1000 per class). TMix is more effective on datasets with more classes, such as yahoo (10 classes), and slightly less on datasets with fewer classes, such

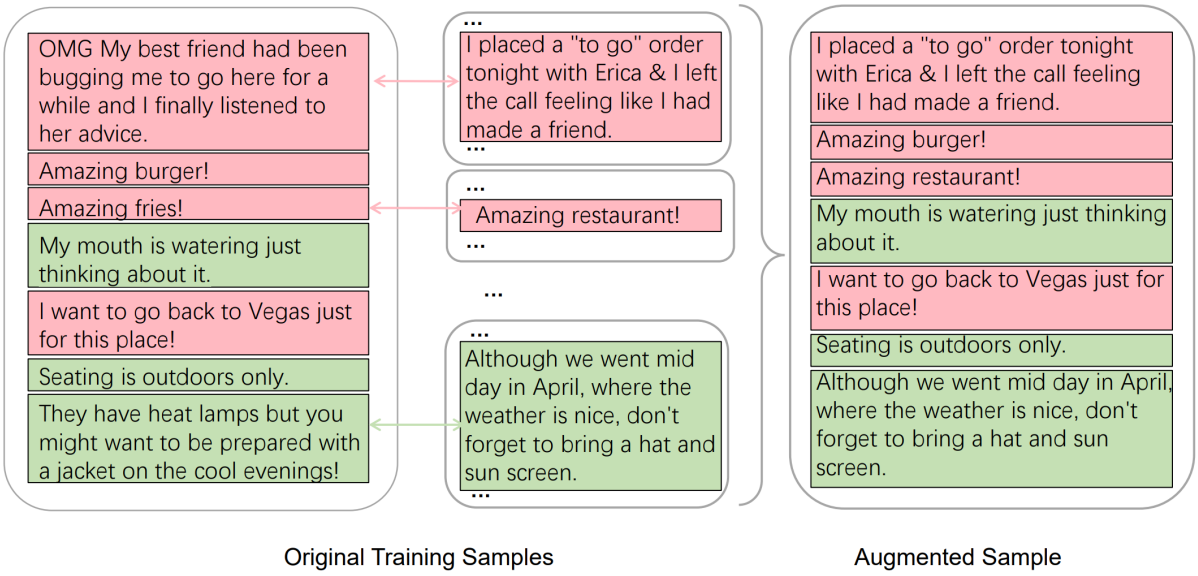

Figure 3: Case study of using SeRe for text augmentation. All samples in the figure are from Yelp-5 and labeled as 4. It is an augmentation between a series of positive reviews. The red sentences are grouped and recombined in the True Bucket, and the green ones are in the False Bucket.

Method	**Accuracy**(%)
SeRe	**72.39**
- Unlabeled	72.15
- Labeled	70.48
- Mixed	71.26
- All	69.28
- Self-training	71.80

Table 4: Accuracy on Yahoo! with removing different parts.

as IMDb (2 classes). Our method does not destroy the semantic and grammatical structure and generates various augmented data through sentence recombination, which is effective on all datasets.

5.4 Ablation Studies

In this section, we perform ablation studies to show the effectiveness of each component in SeRe. As shown in Table 4, we remove each component and show the results. The performance decreased 3.11% after removing all augments, which indicates the proposed SeRe is helpful in semi-supervised text classification tasks. Specifically, in the three types of augmentation of $D_l^{aug}, D_u^{aug}, D_{mix}^{aug}$, the performance after removing the labeled data augmentation results in the most significant degradation of 1.91%. On the other hand, the performance decreased 0.59% after removing the self-training component.

5.5 Case Study

We perform a case study to show the effect of data augmentation with SeRe. As shown in Fig. 3, the example comes from the training data with label 4 in the Yelp-5 (Zhang et al., 2015) dataset. The text contains seven sentences and shows a positive review of a restaurant. According to the augmentation process of SeRe, all sentences are divided into two groups. The red ones represent sentences with high classification confidence, which are assigned to the True Bucket, and the green ones are assigned to the False Bucket. Three randomly selected sentences are replaced in their respective buckets with sentences from other samples with the closest semantic representation. The generated augmented sample is composed of different sentences and maintains the same semantics as the original sample.

6 Conclusion

In this paper, we propose SeRe to improve the feature diversity of training data by grouping and recombining sentences of different samples. We applied SeRe to semi-supervised text classification tasks to obtain state-of-the-art performance. The combinatorial relationship between sentences is focused on in this paper. More fine-grained such as token-level combination relations, need to be further studied.

References

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A. Raffel. 2019. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, volume 32, pages 5049–5059.

Ming-Wei Chang, Lev-Arie Ratinov, Dan Roth, and Vivek Srikumar. 2008. Importance of semantic representation: Dataless classification. In *Aaai*, volume 2, pages 830–835.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Ekin Dogus Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. 2018. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jingfei Du, Edouard Grave, Beliz Gunel, Vishrav Chaudhary, Onur Celebi, Michael Auli, Veselin Stoyanov, and Alexis Conneau. 2021. Self-training improves pre-training for natural language understanding. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5408–5418, Online. Association for Computational Linguistics.

Akbar Karimi, Leonardo Rossi, and Andrea Prati. 2021. Aeda: An easier data augmentation technique for text classification. *arXiv preprint arXiv:2108.13230*.

Alex Kurakin, Colin Raffel, David Berthelot, Ekin Dogus Cubuk, Han Zhang, Kihyuk Sohn, and Nicholas Carlini. 2020. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring.

Dong-Hyun Lee et al. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896.

Joseph Lemley, Shabab Bazrafkan, and Peter Corcoran. 2017. Smart augmentation learning an optimal data augmentation strategy. *IEEE Access*, 5:5858–5869.

Xiaonan Li, Hang Yan, Xipeng Qiu, and Xuanjing Huang. 2020. Flat: Chinese ner using flat-lattice transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6836–6842.

Chen Liu, Mengchao Zhang, Zhibin Fu, Pan Hou, and Yu Li. 2021. Flitext: A faster and lighter semi-supervised text classification with convolution networks. *arXiv preprint arXiv:2110.11869*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Xuan Wang, Yu Zhang, Heng Ji, and Jiawei Han. 2021. Distantly-supervised named entity recognition with noise-robust learning and language model augmented self-training. *arXiv preprint arXiv:2109.05003*.

Takeru Miyato, Andrew M. Dai, and Ian J. Goodfellow. 2017. Adversarial training methods for semi-supervised text classification. In *ICLR (Poster)*.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. 2018. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993.

Luis Perez and Jason Wang. 2017. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Yaqing Wang, Subhabrata Mukherjee, Xiaodong Liu, Jing Gao, Ahmed Hassan Awadallah, and Jianfeng Gao. 2021. List: Lite self-training makes efficient few-shot learners. *arXiv preprint arXiv:2110.06274*.

Jason W. Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. 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 6381–6387.

Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. 2019. Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*.

Diyi Yang, Jiaao Chen, Zichao Yang, Dan Jurafsky, and Eduard H. Hovy. 2019a. Let's make your request more persuasive: Modeling persuasive strategies via semi-supervised neural nets on crowdfunding platforms. 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 3620–3630.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019b. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32, pages 5753–5763.

Zichao Yang, Zhiting Hu, Ruslan Salakhutdinov, and Taylor Berg-Kirkpatrick. 2017. Improved variational autoencoders for text modeling using dilated convolutions. In *ICML'17 Proceedings of the 34th International Conference on Machine Learning - Volume 70*, pages 3881–3890.

Wen Zhang, Yang Feng, Fandong Meng, Di You, and Qun Liu. 2019. Bridging the gap between training and inference for neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4334–4343.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28:649–657.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. Extractive summarization as text matching. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6197–6208.