On-the-fly Denoising for Data Augmentation in Natural Language Understanding

Tianqing Fang[1*], **Wenxuan Zhou**[2], **Fangyu Liu**[3], **Hongming Zhang**[4], **Yangqiu Song**[1], **Muhao Chen**[2,5]

[1]Hong Kong University of Science and Technology [2]University of Southern California
[3]University of Cambridge [4]Tencent AI Lab, Seattle [5]University of California, Davis
{tfangaa, yqsong}@cse.ust.hk, zhouwen@usc.edu, fl399@cam.ac.uk,
hongmzhang@global.tencent.com, muhchen@ucdavis.edu

Abstract

Data Augmentation (DA) is frequently used to provide additional training data without extra human annotation automatically. However, data augmentation may introduce noisy data that impairs training. To guarantee the quality of augmented data, existing methods either assume no noise exists in the augmented data and adopt consistency training or use simple heuristics such as training loss and diversity constraints to filter out "noisy" data. However, those filtered examples may still contain useful information, and dropping them completely causes a loss of supervision signals. In this paper, based on the assumption that the original dataset is cleaner than the augmented data, we propose an on-the-fly denoising technique for data augmentation that learns from soft augmented labels provided by an organic teacher model trained on the cleaner original data. To further prevent overfitting on noisy labels, a simple self-regularization module is applied to force the model prediction to be consistent across two distinct dropouts. Our method can be applied to general augmentation techniques and consistently improve the performance on both text classification and question-answering tasks[1].

1 Introduction

The development of natural language understanding (NLU) comes along with the efforts in curating large-scale human-annotated datasets (Brown et al., 2020; Srivastava et al., 2022). The performance of NLP models usually highly correlates with the quantity and quality of training data. However, human data annotations are usually expensive to acquire and hard to scale (Paulheim, 2018). To address this challenge, automatic data augmentation becomes an attractive approach to effectively increase the scale of training data, and improve the performance of neural models, particularly in low-resource scenarios (Wei and Zou, 2019; Xie et al., 2020a; Yang et al., 2020; Feng et al., 2021).

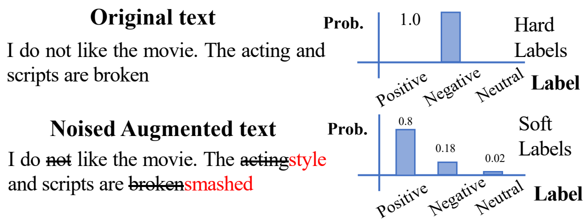

Figure 1: An example in a sentiment classification task about the noise brought by text-editing data augmentation. The noisy augmented text has the probability of being a "positive" attitude due to the removal of "not".

However, automatic data augmentation techniques, regardless of token-level (Wei and Zou, 2019; Xie et al., 2020a) or sentence-level (Sennrich et al., 2016; Yang et al., 2020) ones, may introduce noise to the augmented data. For example, in text classification or sentiment analysis tasks, altering or removing some decisive words can change the original label (Troiano et al., 2020). In addition, automatic data augmentation may distort the core semantic meaning or impair the fluency of the original text, leading to meaningless data instances (Bayer et al., 2021).

To improve the quality of augmented data, various filtering techniques have been developed to select a subset of high-quality data. Typical filtering paradigms design an uncertainty- or diversity-based metric to select data examples, for which the metric could be the loss of the task model trained on the original data (Zhao et al., 2022; Kamalloo et al., 2022), diversity of the augmented data (Zhao et al., 2022; Yang et al., 2020; Kim et al., 2022), influence functions (Yang et al., 2020), and logit consistency across multiple trained models (Li et al., 2020; Zhou et al., 2021). However, data filtering mechanisms set a *discrete* threshold and potentially

[*] Work done when visiting USC.
[1]Our code is available at https://github.com/luka-group/ODDA-Data-Augmentation

Findings of the Association for Computational Linguistics: EACL 2024, pages 766–781
March 17-22, 2024

discard examples that the model can still acquire signals from using properly designed denoising objectives (Li et al., 2020). Alternative solutions to *continuously* re-weighting (Yi et al., 2021) augmented data or adopting consistency training (Xie et al., 2020a) often focus solely on the learnability of data or assume noisy examples should have the same label as the original ones, rather than mitigating their noise.

In this paper, we address the problem of *learning from noisy augmented data* without (1) the effort of producing extra augmentations for filtering and (2) the risk of losing useful supervision signals from examples that are *discretely* filtered out. Noisy data augmentation does not necessarily lead to a hard flipped label but a soft change in the original label distribution, as illustrated in Fig. 1. Therefore, we propose a soft noisy label correction framework called On-the-fly Denoising for Data Augmentation (ODDA), which distills task signals to noisy augmented instances and proactively mitigates noise. Different from the *learning from noisy label* (LNL) setting in fully supervised (Wang et al., 2019a,b; Zhou and Chen, 2021) or distantly supervised training (Meng et al., 2021), in data augmentation, the original dataset is cleaner and offers a natural distributional prior for estimating the noise level of augmented data, since the purpose of training data creation always involves approximating the data distribution in test time. This assumption is also used in other works such as NoisyStudent (Xie et al., 2020b). To leverage such signals, we propose an Organic Distillation[2] module that uses a teacher model finetuned on the cleaner original dataset to provide soft labels for augmented data, where noisy data are softly relabeled to prevent the student model from overfitting to wrong labels. Besides augmentation noise, the original data and organic distillation may also bring the noise. To address this issue, we further add a dropout-enabled self-regularization objective to force the predicted label distributions to be similar across two different dropout masks. It is based on the observations that noisy labels may be forgotten during training or by perturbations, and self-regularization will force the consistency between perturbations and improve noise robustness (Aghajanyan et al., 2021).

To summarize, the contributions of this paper are three-fold. First, we cast light on the problem of learning from noisy augmented data with *soft label correction* instead of discretely filtering them out. Second, we propose a simple yet effective on-the-fly denoising technique that continuously distills useful task signals to noisy augmentations, coupled with a self-regularization loss to reduce overfitting to noise in general. Third, we conduct extensive experiments on two NLU tasks, text classification and question answering, and show the effectiveness of our method for denoising both representative token-level and sentence-level data augmentation techniques.

2 Related Works

Data Augmentation and Filtering Recent studies on data augmentation for NLP have led to two main paradigms: *token-level augmentation* and *sentence-level augmentation* (Chen et al., 2021). Token-level augmentation conduct text editing on tokens from the input text. Such techniques include using synonym replacement (Zhang et al., 2015; Wang and Yang, 2015; Kobayashi, 2018) and word replacement with contextualized embedding or a masked language model (Yi et al., 2021; Kumar et al., 2020), etc. Particularly, EDA (Wei and Zou, 2019) combines paraphrasing and random deletion, insertion, and swapping to perturb the text for augmentation. Sentence-level augmentation, on the other hand, modifies the whole sentence at once. Methods include paraphrase-based augmentation techniques such as back-translation (Sennrich et al., 2016; Yu et al., 2018) and paraphrase generation (Prakash et al., 2016). Another popular approach is to use conditional text generation models finetuned on the task dataset to automatically synthesize more training data. It has been applied to tasks such as text classification (Anaby-Tavor et al., 2020; Kumar et al., 2020), machine reading comprehension (Puri et al., 2020) , relation extraction (Hu et al., 2023), commonsense reasoning (West et al., 2022; Yang et al., 2020), and dialogue systems (Kim et al., 2023). Another line of research operates on the embedding space. MIXUP-related augmentation generates augmented samples based on interpolating word embedding and label embedding vectors (Chen et al., 2020; Si et al., 2021). Instead of focusing on concrete augmentation techniques, our paper study denoising synthetic data provided by any data augmentation method.

[2]We call it *organic* as the teacher model for distillation is trained on the original dataset.

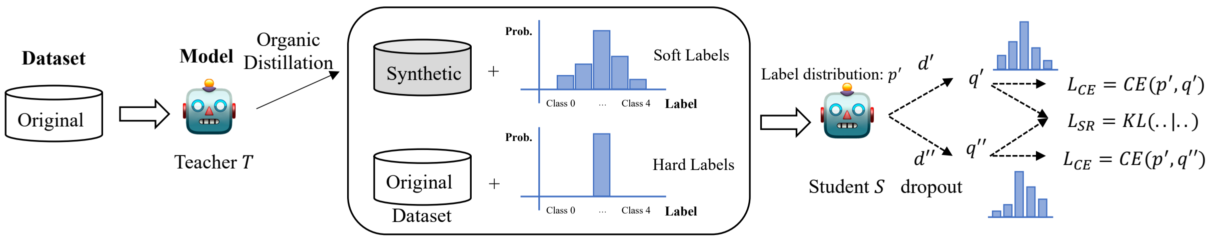

Figure 2: Overview of our ODDA framework.

Learning with Noisy Labels In the field of NLP, particularly in low-resource settings, it is necessary to address the challenge of handling noisy labels derived from inaccurate annotations (Zhou and Chen, 2021), pseudo labels (Li et al., 2020), weak labels (Zeng et al., 2022), augmented data (Kamalloo et al., 2022), and other sources. Various techniques have been developed to combat labeling noise in NLP datasets. Filtering-based techniques identify noisy examples through training dynamics or latent space features and then filter them out to produce a cleaner and more selective training dataset. Such techniques are based on prediction consistency of different models (Zhou et al., 2021), loss-based uncertainty estimation (Han et al., 2018), and feature or representation-based outlier detection (Wu et al., 2020; Feng et al., 2021; Wang et al., 2022a). Besides noise filtering, an alternative approach to learning from noisy labels is to add an auxiliary learning objective to improve the noise robustness of a supervised model. Techniques of this kind include mixing up noisy examples (Zhang et al., 2018), consistency training (Xie et al., 2020a,b), co-regularization (Zhou and Chen, 2021), curriculum loss (Lyu and Tsang, 2020), and semi-supervised training on noisy data (Li et al., 2020).

In data augmentation, recent studies have suggested using a filtering mechanism to select high-quality synthetic data from potentially noisy ones. Typical filters include diversity (Zhao et al., 2022), task loss (Fang et al., 2022), consistency between two models (Wang et al., 2022b), influence function (Yang et al., 2020), similarity with original data (Avigdor et al., 2023), and the alignment of the fully augmented Jacobian with labels/residuals (Liu and Mirzasoleiman, 2022). Instead of filtering, our method continuously learns from noisy labels with a cleaner teacher model and a denoising objective without discarding noisy instances, thus can more sufficiently acquire supervision signals from all augmented instances. Our work also differs from consistency training, which assumes that augmented data, even if noisy, should have similar predictions to the original instances. In contrast, we aim to mitigate such noise, which runs counter to the objective of consistency training.

3 Method

This section introduces the problem formulation (§3.1) and our ODDA framework (§3.2-§3.3).

3.1 Problem Formulation

We consider the problem formulation of general text classification tasks. We denote the dataset as $\mathcal{D} = \{(x_i, y_i)\}, i = 1, \cdots, n$, where x_i is the input text, $y_i \in \mathcal{Y}$ is the label of x_i from the pre-defined label set $\mathcal{Y}$, and n is the number of instances in the dataset. A data augmentation algorithm derives an augmented dataset $\mathcal{D}' = \{(x'_i, y'_i)\}, i = 1, \cdots, kn$ from the original dataset $\mathcal{D}$, with an amplification factor k denoting that for each data instance we generate k augmentations. We use both the original dataset $\mathcal{D}$ and the augmented dataset $\mathcal{D}'$ to train the classifier. Other NLU tasks, such as sentiment analysis, multiple-choice question answering, and natural language inference, can be easily converted to a text classification paradigm. For example, multiple-choice question answering can be converted to text classification by treating each question-answer pair as an input instance.

3.2 On-the-fly Denoising

This subsection introduces the details of our On-the-fly Denoising for Data Augmentation (ODDA) framework. ODDA first trains an (organic) teacher model on the original dataset and then uses this teacher model to assign soft labels to the augmented dataset. During the learning process of augmented data, the model is jointly trained with two denoising objectives, where one is a cross-entropy loss on the distilled soft labels, and the other is a self-regularization loss to encourage robustness and consistency across two different dropout masks to automatically correct the noisy labels. The latter is important as the teacher model may also bring the noise to the soft labels, and self-regularization can serve as a general denoising channel for both forms of noise. An overview illustration of ODDA is shown in Fig. 2.

Organic Distillation (OD). The first component of our framework is Organic Distillation. We first train a teacher model on the original training dataset D. The resulting model (the *organic teacher*), denoted as T, uses the same model architecture as the later student model. Denote $z = f_T(x)$ as the function that produces logits z given input x using the teacher model T. For an instance x, the teacher model can predict the soft probability over the label set $\mathcal{Y}$ with a temperature-controlled softmax $g(z, \tau)$:

$$q_y = g(z, \tau)_y = \frac{\exp\left(z_y/\tau\right)}{\sum_{j \in \mathcal{Y}} \exp\left(z_j/\tau\right)}, \quad (1)$$

where q_y is a predicted probability of a class y from $\mathcal{Y}$, τ is a temperature hyperparameter where a larger temperature results in a smoother distribution. Specifically, we omit $\tau = 1$ in $g(\cdot, \tau)$, and use $g(x)$ to represent the standard softmax function. We denote $f(x)$ as the student model that produces logits, and the loss function as cross-entropy loss $l_{\text{CE}}(p, q) = -(q \log p + (1-q) \log(1-p))$, where p denotes the ground labels and q denotes the predicted probabilities.

Organic distillation distills knowledge from the organic teacher model to the augmented data. As the original dataset is inherently of better quality than the augmented data, it can be used to provide a distributional prior on the level of noisiness in augmented data, thus calibrating the learning process of data augmentation and preventing overfitting the labeling noise. For an augmented data instance (x', y'), we first compute the soft probabilities predicted by the organic teacher as $q' = g(f_T(x'), \tau)$, as in equation (1). Then $p' = g(f(x'))$ is the probability distribution over the label set $\mathcal{Y}$ predicted by the student model when training on synthetic data. Then the corresponding loss function of organic distillation on the augmented example x' is:

$$\begin{aligned} \mathcal{L}_{\text{OD}}(x') &= l_{\text{CE}}(p', q') \\ &= l_{\text{CE}}\Big(g\big(f(x')\big), g\big(f_T(x'), \tau\big)\Big). \end{aligned} \quad (2)$$

Algorithm 1 On-the-fly DA Denoising (ODDA)

Input: Teacher model $f_T(\cdot)$, student model $f(\cdot)$, original dataset $\mathcal{D} = \{(x_i, y_i)\}, i = 1, \cdots, n$, augmented dataset $\mathcal{D}' = \{(x'_i, y'_i)\}, i = 1, \cdots, kn$, OD temperature τ, SR coefficient α. Max training steps for the organic teacher s_T and the student s_S.

Output: The trained student model $f(\cdot)$

```
Initialize the teacher model f_T(·)
s ← 0                                        ▷ Training steps for OD
while s < s_T do
    Sample a batch B from {(x_i, y_i)}
    Train f_T(·) with cross-entropy loss on B
end while
s ← 0                                  ▷ Training steps for Denoising
D⁺ ← {(x_i, y_i)} ∪ {(x'_i, y'_i)}                  ▷ Mix D & D'
while s < s_S do
    Sample a batch B' from D⁺
    Train f(·) with loss in Eq. (4) on B' with Organic
    Distillation and Self-Regularization to do deonising
end while
```

Self-Regularization (SR). As the OD module may also introduce noise to the learning process, we introduce another general denoising channel. Recent studies have shown that noisy instances generally tend not to be "memorized" easily by machine learning models, and are frequently "forgetten" given small perturbations (Xie et al., 2020a; Aghajanyan et al., 2021) and along with the training steps (Zhou and Chen, 2021). The often inconsistent characteristics of noisy instances over the learning curve is mainly attributed to their contradiction to the model's overall task inductive bias represented coherently by the clean data. To mitigate the impact of noise from individual data instances, inconsistent outputs resulting from small perturbations should be corrected." Instead of filtering noisy examples out with the risk of losing useful information, we learn from noisy (and clean) examples with an additional objective by bounding the model's output to be consistent under small perturbations. Following R-Drop (Liang et al., 2021), the perturbations are introduced with dropout, and a regularization loss forcing the model prediction to be consistent across two different dropout outputs is adopted[3]. Denote $d(f(x))$ as the function that outputs the predicted probability distribution under a dropout mask d, and d_i is the i-th dropout mask. Then the self-regularization loss is defined as the Kullback-Leibler (KL) divergence between the average probability distribution of the m dropout operations and the output of each dropout:

[3]A detailed explanation and theoretical analysis to self-regularization is presented in Appx. §B.

$$\bar{p} = \frac{1}{m}\sum_{i=1}^{m} g(d_i(f(x'))),$$

$$\mathcal{L}_{\text{SR}}(x') = \frac{1}{m}\sum_{i=1}^{m} \text{KL}\Big(\bar{p}||g\big(d_i(f(x'))\big)\Big). \quad (3)$$

3.3 Joint Training

In the end, the model is jointly trained with the **OD** and **SR** objectives on the original dataset $\{(x_i, y_i)\}$ and the augmented dataset $\{(x'_i, y'_i)\}$:

$$\begin{aligned}\mathcal{L} =& \frac{1}{n}\sum_{i=1}^{n} l_{\text{CE}}\Big(g\big(f(x_i)\big), y_i\Big) \\ &+ \frac{1}{kn}\sum_{i=1}^{kn} \mathcal{L}_{\text{OD}}(x'_i) \\ &+ \alpha\frac{1}{kn+n}\sum_{i=1}^{kn+n} \mathcal{L}_{\text{SR}}(x'_i). \quad (4)\end{aligned}$$

The overall loss function is the sum of the cross-entropy loss on the original data with hard labels, the cross-entropy loss of the augmented data with soft labels distilled with the organic teacher, and the KL divergence between the average probability across m different dropouts and each of the m dropouts. Here $l_{\text{CE}}(\cdot)$ is the cross-entropy loss function, n is the number of original examples and k is the amplification factor for data augmentation, and α is a hyper-parameter to control the effect of self-regularization. In the third term, the SR is applied to both the original and augmented data, where the number of instances $n + kn$ indicates the collection of both the original and augmented data. Though we derive these formulations based on the text classification task, in multiple-choice QA tasks, the formulation can be accordingly converted to a c-class classification task, where c is the number of choices per question. The algorithm is outlined in Alg. 1.

4 Experiments

This section introduces experimental settings and results analysis. We evaluate on two representative tasks in NLU, few-shot text classification (Section §4.1) and multiple-choice (commonsense) question answering (Section §4.2). We use EDA (Wei and Zou, 2019) as a representative token-level based augmentation method for text classification, and use Generative Data Augmentation (G-DAUG) (Yang et al., 2020) to explore task-aware sentence-level augmentation methods for hard QA tasks that require commonsense reasoning abilities. In Section §4.3, we provide ablation studies to show the effect of ODDA under synthetic noise on augmented data, the influence of hyperparameters, and the effect of denoising modules.

4.1 Text Classification

Setup. Following the previous work (Zhao et al., 2022), we use five text classification datasets: **TREC** (Li and Roth, 2002) (Question classification, n=5,452), **Irony** (Hee et al., 2018) (Tweets Irony Classification, n=3,817), **AGNews** (Zhang et al., 2015) (News Classification, n=120,000), **Sentiment** (Rosenthal et al., 2017) (Tweets Sentiment Analysis, n=20,631), and **Offense** (Founta et al., 2018) (Tweets Offense Detection, n=99,603). We randomly sample different proportions of each dataset for experiments to fully demonstrate the effect of data augmentation, where the percentage in Tab. 1 (%) indicates the percentage of data sampled for training, leading to around 100 and 1000 examples sampled for the two few-shot proportions, respectively. BERT-base (Devlin et al., 2019) is used as the backbone model for all the text classification experiments, which is incorporated with EDA (Wei and Zou, 2019) for data augmentation. The augmentation probability of the four edit operations in EDA is equally set as 0.05. We report the average macro-F1 across five different random seeds and the standard deviation in subscripts. Each original data example is associated with $k = 3$ augmented data. The OD temperature τ is searched within {0.5, 1, 2, 3}, and the SR α is searched within {5, 10, 20, 50, 100}. Early stopping is used to select the model with the best performance. More hyperparameters are shown in Appx. §A.1.

Baselines. We compare three types of baseline denoising techniques, which are filtering, reweighting, and consistency training. For filtering, we use EPiDA (Relative Entropy Maximization + Conditional Entropy Minimization, Zhao et al. (2022)), Glitter (selecting augmented data with higher task loss, Kamalloo et al. (2022)), Large-loss (select augmented data with small loss, Han et al. (2018)), to filter out low-quality augmented training data. For re-weighting, we use the re-weighting factors in Yi et al. (2021), where examples with larger training loss are given larger weights. For consistency training (denoted as Consist.), we use the idea in Unsupervised Data Aug-

Method	TREC		Irony		AGNews		Sentiment		Offense	
	1%	10%	1%	10%	0.05%	0.1%	1%	10%	0.1%	1%
Sup.	60.64$_{\pm0.60}$	90.53$_{\pm0.47}$	55.48$_{\pm1.05}$	63.14$_{\pm0.99}$	84.05$_{\pm0.47}$	86.43$_{\pm0.07}$	54.10$_{\pm1.22}$	65.56$_{\pm0.22}$	51.91$_{\pm0.53}$	64.35$_{\pm0.12}$
Data Augmentation										
EDA	61.68$_{\pm0.29}$	93.83$_{\pm0.63}$	57.07$_{\pm0.66}$	64.55$_{\pm0.52}$	84.01$_{\pm0.18}$	86.43$_{\pm0.07}$	56.57$_{\pm0.75}$	65.80$_{\pm0.14}$	51.86$_{\pm0.37}$	64.61$_{\pm0.15}$
EPiDA	64.92$_{\pm0.50}$	93.96$_{\pm0.18}$	58.25$_{\pm0.95}$	64.72$_{\pm0.58}$	84.51$_{\pm0.31}$	86.68$_{\pm0.19}$	57.20$_{\pm0.32}$	65.58$_{\pm0.24}$	51.55$_{\pm0.49}$	64.45$_{\pm0.16}$
Glitter	64.16$_{\pm0.20}$	93.55$_{\pm0.06}$	58.76$_{\pm0.44}$	64.73$_{\pm0.95}$	84.84$_{\pm0.32}$	87.00$_{\pm0.29}$	**57.73**$_{\pm0.31}$	65.52$_{\pm0.20}$	51.69$_{\pm0.42}$	64.45$_{\pm0.15}$
Large-loss	62.21$_{\pm1.71}$	94.06$_{\pm1.90}$	57.07$_{\pm2.13}$	64.42$_{\pm1.28}$	83.48$_{\pm0.97}$	86.43$_{\pm0.28}$	57.13$_{\pm1.27}$	65.66$_{\pm0.49}$	51.78$_{\pm0.77}$	64.49$_{\pm0.41}$
Re-weight	64.37$_{\pm1.69}$	95.28$_{\pm0.97}$	58.14$_{\pm2.34}$	64.56$_{\pm1.73}$	84.45$_{\pm1.12}$	86.82$_{\pm0.50}$	56.81$_{\pm1.52}$	65.55$_{\pm1.50}$	51.70$_{\pm1.10}$	64.54$_{\pm0.43}$
Consist.	65.55$_{\pm0.81}$	95.15$_{\pm0.90}$	58.32$_{\pm1.71}$	64.50$_{\pm1.24}$	84.34$_{\pm0.78}$	86.45$_{\pm0.26}$	57.10$_{\pm1.26}$	65.64$_{\pm0.46}$	51.86$_{\pm0.98}$	64.66$_{\pm0.43}$
Denoising Data Augmentation (EDA as the DA algorithm)										
Ours (OD)	65.17$_{\pm1.25}$	95.02$_{\pm1.42}$	58.51$_{\pm2.67}$	64.73$_{\pm0.18}$	84.91$_{\pm0.44}$	86.84$_{\pm0.26}$	57.09$_{\pm1.63}$	65.68$_{\pm0.51}$	52.13$_{\pm1.43}$	65.16$_{\pm0.64}$
Ours (SR)	65.87$_{\pm1.22}$	95.50$_{\pm0.68}$	57.51$_{\pm1.92}$	64.24$_{\pm0.61}$	84.80$_{\pm0.57}$	86.75$_{\pm0.57}$	57.42$_{\pm1.09}$	65.74$_{\pm0.27}$	52.01$_{\pm0.99}$	65.06$_{\pm0.49}$
Ours (both)	**67.16**$_{\pm0.37}$	**96.04**$_{\pm0.08}$	**60.66**$_{\pm1.43}$	**65.54**$_{\pm0.37}$	**86.30**$_{\pm0.13}$	**87.14**$_{\pm0.17}$	57.17$_{\pm0.37}$	**65.90**$_{\pm0.19}$	**52.34**$_{\pm0.53}$	**65.43**$_{\pm0.29}$

Table 1: Performance of different filtering and re-weighting methods on the five text classification datasets, where EDA is used as the base data augmentation algorithm for all methods. 1% means using 1% of the original training data for training. We report the average f1 score across five different random seeds.

mentation (UDA; Xie et al., 2020a) to add a consistency loss between original examples and the corresponding augmented examples. More details are provided in Appx. §A.1.

Results and Analysis. The main experimental results of text classification are presented in Tab. 1. First, we can see that ODDA can provide remarkable improvements over EDA, the base data augmentation method without any filtering or denoising. The notable improvement of F1 2.5% increase in average for the smaller few-shot split and 1.0% F1 increase in average for the larger few-shot split over EDA indicate the importance of addressing the noise issue in augmented data.

Second, ODDA outperforms filtering-based baselines (EPiDA, Glitter, and Large-loss) in all datasets and splits except for the 1% Sentiment. Note that these baselines need to select $k = 3$ augmented examples per original example from a candidate pool of 50 EDA-generated augmented examples per original example, while in our method directly generates the $k = 3$ augmented examples per original instance. Those filtering baselines are more costly and require generating 16 times more augmentations than our method to perform filtering. We can conclude that learning with a denoising objective for data augmentation can be far more data efficient than filtering by exploiting the denoising training signals from noisy examples without filtering them out.

Third, ODDA outperforms re-weighting and Consist. by a large margin. These two methods adopt an opposite idea of denoising to some extent. For re-weighting, augmented examples with larger training loss, which can be regarded as more noisy (Shu et al., 2019), will be up-weighted during training, while in our Organic Distillation and Sefl-regularization, examples identified noisier will be down-weighted to rectify the effect of noisy augmented instances. For Consistency training, it assumes that the original and its corresponding augmented example should share the same label and train them with a consistency loss, which is also opposite to our assumption that augmented data may be noisy. From the comparison of those two methods, we can conclude that the denoising objective better suits the scenario of data augmentation than both the learnability-based re-weighting and the consistency training with label-preserving assumption.

4.2 Commonsense Question Answering

Setup. We follow the setups in G-DAUG (Yang et al., 2020) to conduct commonsense QA experiments. We study a full-shot setting here for the QA tasks as a supplement to the few-shot text classification experiments, and select two representative multiple-choice commonsense QA datasets, WinoGrande (Sakaguchi et al., 2020) and CommonsenseQA (CSQA; Talmor et al. 2019). Other datasets are not selected as they either adopt a few-shot setting, or the augmented data is not publicly available. We use the released version of augmented data by Yang et al. (2020)[4] produced with finetuned GPT-2 (Radford et al., 2019).

[4]https://github.com/yangyiben/G-DAUG-c-Generative-Data-Augmentation-for-Commonsense-Reasoning

	WinoGrande						CSQA
	XS	S	M	L	XL	AUC	
Supervised	60.28$\pm$1.52	62.23$\pm$2.06	66.00$\pm$1.28	74.68$\pm$0.28	79.09$\pm$0.56	68.12	76.35$\pm$0.31
G-DAUG	60.49$\pm$0.44	66.04$\pm$0.48	72.22$\pm$0.43	76.79$\pm$0.77	80.09$\pm$0.53	71.32	77.38$\pm$0.36
Ours (OD)	61.18$\pm$0.59	67.45$\pm$0.47	72.38$\pm$0.73	77.35$\pm$0.22	80.75$\pm$0.36	72.01	78.41$\pm$0.40
Ours (SR)	60.68$\pm$0.72	67.06$\pm$0.69	72.34$\pm$0.68	77.09$\pm$0.38	80.57$\pm$0.56	71.76	77.62$\pm$0.41
Ours (both)	**61.30**$\pm$0.55	**67.62**$\pm$0.48	**72.68**$\pm$0.70	**77.65**$\pm$0.21	**80.80**$\pm$0.51	**72.23**	**78.69**$\pm$0.31

Table 2: Performance of commonsense question answering.

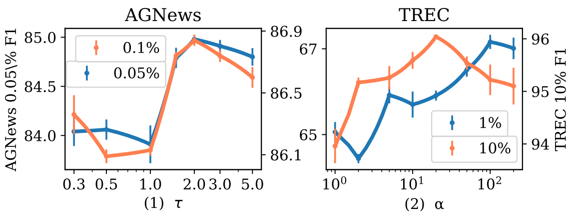

Figure 3: (1) The effect of OD temperature τ on the classification performance for AGNews dataset. (2) The effect of SR coefficient α on the classification performance for TREC dataset.

RoBERTa-large (Liu et al., 2019) is used as the backbone QA model, and the hyperparameters are the same as in Yang et al. (2020). We evaluate the model performance using accuracy for each subset in WinoGrande, and an AUC calculated with the curve of the logarithm of the number of instances of each subset against the corresponding accuracy, to present an overall performance on WinoGrande across the five subsets. Accuracy is used for CSQA as the evaluation metric. As linear learning rate decay is applied during the training, we report the performance of the last checkpoint during training. Different from the original paper of G-DAUG (Yang et al., 2020), which reports the performance of only one run, we report the average and standard deviation across five different random seeds. More details about models and datasets are presented in Appx. §A.2.

Baselines. As in G-DAUG, the augmented instances are already filtered with an influence function (Koh and Liang, 2017) and diversity heuristics, we do not conduct further filtering as baselines. And as no direct mapping exists between the original and augmented examples, the re-weighting and consistency training baseline does not fit the sentence-level data augmentation setting. Hence, we only compare the performance of adding our on-the-fly denoising technique on top of the already-filtered augmented dataset against the performance of G-DAUG and the supervised learning baseline without data augmentation. We also check the effect of each channel (OD and SR).

Results and Analysis. The QA results are shown in Tab. 2. When we apply ODDA to the augmented data generated by G-DAUG filtered with influence function and a diversity heuristic defined in Yang et al. (2020), the performance can be consistently improved across different few-shot splits of WinoGrande and full-shot CSQA. These experiments first demonstrate that besides token-level data augmentation, where each augmented example can be aligned with its original example, ODDA can also work well for sentence-level data augmentation, where there is no explicit mapping between augmented data and original data. This is an advantage as some data augmentation boosting methods need to leverage the mapping between original and augmented examples to select semantically similar augmentations (e.g., EPiDA) or use consistency training, while our method is not restricted by this precondition. Second, we show that our method can not only be used for boosting text classification, but can work well for more complex commonsense reasoning tasks.

4.3 Ablation Study

Organic teacher distillation. The Organic Distillation (OD) module distills the knowledge from the relatively cleaner original dataset to the augmented data with soft labels, preventing overfitting on hard noisy labels. We check the influence of the distillation temperature τ on the model performance, shown in Fig. 3 (1) for the AGNews dataset as an example. Specifically, the model performance reaches its best when the temperature $\tau = 2$, indicates a softer label distribution. For other datasets such as TREC, Irony, and Offense, the variance of different temperatures is relatively minor, and we select $\tau = 1$ as the default. While for AGNews and Sentiment, the model can benefit from larger temperature, which may indicate that there is more noise in the augmented data from those

Method	Irony 10%			
	$p_n = 0.0$	$p_n = 0.1$	$p_n = 0.3$	$p_n = 0.5$
EDA	64.55	63.27	63.26	60.41
EPiDA	64.72	64.57	63.94	63.24
Glitter	64.73	65.04	62.99	61.85
Large-loss	64.42	63.42	63.27	61.56
Re-weight	64.56	64.38	64.53	63.79
Ours (both)	**65.54**	**65.54**	**65.54**	**65.54**

Table 3: Experiments on adding synthetic noise to augmented data for the Irony dataset (10%), when original data remain still. p_n indicates the probability that the label of an augmented example is flipped. As our method learns with the soft labels provided by the clean original dataset, it is not affected by noise on labels in the augmented dataset.

Method	TREC		Irony		AGNews	
	1%	10%	1%	10%	0.05%	0.1%
Iter. Teacher	66.89	95.56	58.73	64.49	84.15	86.17
EMA	64.10	95.26	57.37	64.40	84.16	86.36
Co-Reg	65.19	95.08	58.29	64.86	84.81	86.54
Co-Teaching	64.62	94.69	57.39	65.51	84.83	86.91
Ours (SRx3)	66.19	95.54	58.31	64.56	84.44	86.56
Ours (SRx4)	65.88	95.69	58.95	64.62	84.67	86.33
Ours (OD)	65.17	95.02	58.51	64.73	84.91	86.84
Ours (SR)	65.87	95.50	57.51	64.24	84.80	86.75
Ours (both)	**67.16**	**96.04**	**60.66**	**65.54**	**86.30**	**87.14**

Table 4: Ablations on the effect of Organic Distillation (OD) and Self-Regularization (SR), compared to their counterparts. SRxn means dropouts are done n times.

two datasets, and softer distribution help reduce overfitting on the augmented data.

Self-regularization. The self-regularization (SR) module in our framework serves as a general denoising channel to minimize the discrepancy of model outputs between two dropouts. The α in Equation (4) is the hyperparameter measuring the importance of the denoising effect. We take the TREC dataset as an example to show the effect of α on the model performance as in Fig. 3 (2). We can see that for TREC 1%, the performance reaches the maximum when $\alpha = 100$, and for TREC 10%, the model performs the best when $\alpha = 20$. Such a difference indicates that in TREC 1%, which contains only fewer than 100 training examples, it can benefit more when the effect of self-regularization out-weight the original cross-entropy loss. Similar results are shown in other datasets under the smaller few-shot training set.

Adding synthetic noise. We further show the effect of our denoising method by introducing synthetic noise of different levels to augmented data. The original dataset remains unchanged to show the effect of a cleaner original dataset. To better demonstrate the effect of denoising in augmented data, we control the noise level by setting a probability p_n of flipping the label of augmented data. We select the dataset Irony (with 10% training data) as an example, as Irony is a binary classification task and flipping the label will definitely lead to an opposite label (for other datasets such as AGNews, there may be slight overlaps between different labels). The results are presented in Tab. 3. We can see that EDA and all filtering methods suffer from performance degradation along with increased noise proportions, while our method is not influenced by such synthetic noise as we do not rely on the hard label of augmented data but the soft label provided by the organic teacher model. The performance degradation is not too drastic when p_n increases as the labels of original data are retained. Such an experiment further consolidates the effectiveness of our denoising method for data augmentation.

Alternative denoising techniques. We also study the alternative solutions to our denoising framework. There are alternative ways to the organic teacher. For example, we could iteratively select the best-performed teacher model during the training with augmented data (denoted as an iterative teacher). For the general denoising channel SR, there are other techniques that perform denoising, such as using Exponential Moving Average (EMA) over training steps (Tarvainen and Valpola, 2017), or using the consistency of two independently-trained models to perform logits regularization (Zhou and Chen, 2021). We also study whether increasing the number of dropouts m to do regularization will help the model performance. These experiments are collectively presented in Tab. 4. We can see that our proposed method achieves the best among other alternative choices. For the Iterative Teacher, though the teacher model is iteratively updated, it may lose the information by cleaner original dataset when further trained on the augmented data. For Co-Regularization, it achieves similar performance when two identical models are simultaneously trained to improve consistency. However, it doubles the cost of training. When doing multiple dropouts in self-regularization, the performance on the 1% split of TREC and Irony can be improved when $m > 2$, while for others, the improvements are not significant. Considering that using $m = 3$ or 4 will lead to 1.5 and 2 times the computational cost, we choose $m = 2$ to make the training more efficient while keeping competitive results.

5 Conclusion

In this paper, we address the problem of improving data augmentation via denoising, and propose an efficient on-the-fly data augmentation denoising framework that leverages a teacher model trained on the cleaner original dataset for soft label correction and a self-regularized denoising loss for general denoising. Such a denoising pipeline can well benefit the tasks with limited annotated data and noisy augmented data. Experiments show that our denoising framework performs consistently better than the baselines of filtering, re-weighting, and consistency training, with both token-level and sentence-level data augmentation methods on few-shot text classification and commonsense question-answering tasks.

Acknowledgement

Tianqing Fang was supported by the Hong Kong PhD Fellowship Scheme. Wenxuan Zhou and Muhao Chen were supported by the NSF Grants IIS 2105329 and ITE 2333736, an Amazon Research Award and a Cisco Research Award. Yangqiu Song was supported by the NSFC Fund (U20B2053) from the NSFC of China, the RIF (R6020-19 and R6021-20) and the GRF (16211520 and 16205322) from RGC of Hong Kong. Yangqiu Song thanks the support from the UGC Research Matching Grants (RMGS20EG01-D, RMGS20CR11, RMGS20CR12, RMGS20EG19, RMGS20EG21, RMGS23CR05, RMGS23EG08). Tianqing Fang and Yangqiu Song also thank the support from Tencent AI lab.

Limitations

We only include one representative token-level and sentence-level data augmentation technique in our experiments, while cannot enumerate all others such as masked language models replacing (Yi et al., 2021). In addition, we only include two representative NLU tasks in the experiments while others such as natural language inference (Bowman et al., 2015) are missing due to the limited presentation space. As for the method ODDA itself, we conduct denoising using the training information within a single training step without considering longer dependencies and training dynamics across different training steps or epochs, which can be a future work of this study.

References

Armen Aghajanyan, Akshat Shrivastava, Anchit Gupta, Naman Goyal, Luke Zettlemoyer, and Sonal Gupta. 2021. Better fine-tuning by reducing representational collapse. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Ateret Anaby-Tavor, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, Naama Tepper, and Naama Zwerdling. 2020. Do not have enough data? deep learning to the rescue! In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7383–7390. AAAI Press.

Noa Avigdor, Guy Horowitz, Ariel Raviv, and Stav Yanovsky Daye. 2023. Consistent text categorization using data augmentation in e-commerce. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 313–321, Toronto, Canada. Association for Computational Linguistics.

Markus Bayer, Marc-André Kaufhold, and Christian Reuter. 2021. A survey on data augmentation for text classification. *ACM Computing Surveys*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 632–642. The Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Jiaao Chen, Derek Tam, Colin Raffel, Mohit Bansal, and Diyi Yang. 2021. An empirical survey of data augmentation for limited data learning in NLP. *CoRR*, abs/2106.07499.

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, ACL 2020, Online, July 5-10, 2020*, pages 2147–2157. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Tianqing Fang, Quyet V. Do, Hongming Zhang, Yangqiu Song, Ginny Y. Wong, and Simon See. 2022. Pseudoreasoner: Leveraging pseudo labels for commonsense knowledge base population. *CoRR*, abs/2210.07988.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard H. Hovy. 2021. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 968–988. Association for Computational Linguistics.

Antigoni-Maria Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of twitter abusive behavior. In *Proceedings of the Twelfth International Conference on Web and Social Media, ICWSM 2018, Stanford, California, USA, June 25-28, 2018*, pages 491–500. AAAI Press.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor W. Tsang, and Masashi Sugiyama. 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 8536–8546.

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018. Semeval-2018 task 3: Irony detection in english tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2018, New Orleans, Louisiana, USA, June 5-6, 2018*, pages 39–50. Association for Computational Linguistics.

Xuming Hu, Aiwei Liu, Zeqi Tan, Xin Zhang, Chenwei Zhang, Irwin King, and Philip S. Yu. 2023. GDA: Generative data augmentation techniques for relation extraction tasks. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10221–10234, Toronto, Canada. Association for Computational Linguistics.

Ehsan Kamalloo, Mehdi Rezagholizadeh, and Ali Ghodsi. 2022. When chosen wisely, more data is what you need: A universal sample-efficient strategy for data augmentation. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1048–1062. Association for Computational Linguistics.

Hyunwoo Kim, Jack Hessel, Liwei Jiang, Ximing Lu, Youngjae Yu, Pei Zhou, Ronan Le Bras, Malihe Alikhani, Gunhee Kim, Maarten Sap, et al. 2023. Soda: Million-scale dialogue distillation with social commonsense contextualization. *arXiv preprint arXiv:2212.10465*.

Jaehyung Kim, Dongyeop Kang, Sungsoo Ahn, and Jinwoo Shin. 2022. What makes better augmentation strategies? augment difficult but not too different. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Sosuke Kobayashi. 2018. Contextual augmentation: Data augmentation by words with paradigmatic relations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 452–457. Association for Computational Linguistics.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1885–1894. PMLR.

Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. Data augmentation using pre-trained transformer models. *CoRR*, abs/2003.02245.

Junnan Li, Richard Socher, and Steven C. H. Hoi. 2020. Dividemix: Learning with noisy labels as semi-supervised learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *19th International Conference on Computational Linguistics, COLING 2002, Howard International House and Academia Sinica, Taipei, Taiwan, August 24 - September 1, 2002*.

Xiaobo Liang, Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, and Tie-Yan Liu. 2021. R-drop: Regularized dropout for neural networks. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 10890–10905.

Tian Yu Liu and Baharan Mirzasoleiman. 2022. Data-efficient augmentation for training neural networks. *CoRR*, abs/2210.08363.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Yueming Lyu and Ivor W. Tsang. 2020. Curriculum loss: Robust learning and generalization against label corruption. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

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. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 10367–10378. Association for Computational Linguistics.

Razvan Pascanu and Yoshua Bengio. 2014. Revisiting natural gradient for deep networks. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Heiko Paulheim. 2018. How much is a triple? estimating the cost of knowledge graph creation. In *ISWC*.

Aaditya Prakash, Sadid A. Hasan, Kathy Lee, Vivek V. Datla, Ashequl Qadir, Joey Liu, and Oladimeji Farri. 2016. Neural paraphrase generation with stacked residual LSTM networks. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 2923–2934. ACL.

Raul Puri, Ryan Spring, Mohammad Shoeybi, Mostofa Patwary, and Bryan Catanzaro. 2020. Training question answering models from synthetic data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 5811–5826. Association for Computational Linguistics.

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

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. Semeval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*, pages 502–518. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8732–8740. AAAI Press.

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 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. 2019. Meta-weight-net: Learning an explicit mapping for sample weighting. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 1917–1928.

Chenglei Si, Zhengyan Zhang, Fanchao Qi, Zhiyuan Liu, Yasheng Wang, Qun Liu, and Maosong Sun. 2021. Better robustness by more coverage: Adversarial and mixup data augmentation for robust finetuning. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 1569–1576. Association for Computational Linguistics.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 4444–4451. AAAI Press.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ameet Rahane, Anantharaman S. Iyer, Anders Andreassen, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew K. Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakas, and et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *CoRR*, abs/2206.04615.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4149–4158. Association for Computational Linguistics.

Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1195–1204.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J. Gordon. 2019. An empirical study of example forgetting during deep neural network learning. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Enrica Troiano, Roman Klinger, and Sebastian Padó. 2020. Lost in back-translation: Emotion preservation in neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 4340–4354. International Committee on Computational Linguistics.

Hao Wang, Bing Liu, Chaozhuo Li, Yan Yang, and Tianrui Li. 2019a. Learning with noisy labels for sentence-level sentiment classification. 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 6286–6292, Hong Kong, China. Association for Computational Linguistics.

William Yang Wang and Diyi Yang. 2015. That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using #petpeeve tweets. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 2557–2563. The Association for Computational Linguistics.

Yikai Wang, Xinwei Sun, and Yanwei Fu. 2022a. Scalable penalized regression for noise detection in learning with noisy labels. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 346–355. IEEE.

Yufei Wang, Can Xu, Qingfeng Sun, Huang Hu, Chongyang Tao, Xiubo Geng, and Daxin Jiang. 2022b. Promda: Prompt-based data augmentation for low-resource NLU tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 4242–4255. Association for Computational Linguistics.

Zihan Wang, Jingbo Shang, Liyuan Liu, Lihao Lu, Jiacheng Liu, and Jiawei Han. 2019b. CrossWeigh: Training named entity tagger from imperfect annotations. 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 5154–5163, Hong Kong, China. Association for Computational Linguistics.

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 2019, Hong Kong, China, November 3-7, 2019*, pages 6381–6387. Association for Computational Linguistics.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena D. Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2022. Symbolic knowledge distillation: from general language models to commonsense models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 4602–4625. Association for Computational Linguistics.

Pengxiang Wu, Songzhu Zheng, Mayank Goswami, Dimitris N. Metaxas, and Chao Chen. 2020. A topological filter for learning with label noise. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Qizhe Xie, Zihang Dai, Eduard H. Hovy, Thang Luong, and Quoc Le. 2020a. Unsupervised data augmentation for consistency training. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Qizhe Xie, Minh-Thang Luong, Eduard H. Hovy, and Quoc V. Le. 2020b. Self-training with noisy student improves imagenet classification. In *2020 IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, pages 10684–10695. Computer Vision Foundation / IEEE.

Yiben Yang, Chaitanya Malaviya, Jared Fernandez, Swabha Swayamdipta, Ronan Le Bras, Ji-Ping Wang, Chandra Bhagavatula, Yejin Choi, and Doug Downey. 2020. G-daug: Generative data augmentation for commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 1008–1025. Association for Computational Linguistics.

Mingyang Yi, Lu Hou, Lifeng Shang, Xin Jiang, Qun Liu, and Zhi-Ming Ma. 2021. Reweighting augmented samples by minimizing the maximal expected loss. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Ziqian Zeng, Weimin Ni, Tianqing Fang, Xiang Li, Xinran Zhao, and Yangqiu Song. 2022. Weakly supervised text classification using supervision signals from a language model. In *Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 2295–2305. Association for Computational Linguistics.

Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

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 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 649–657.

Minyi Zhao, Lu Zhang, Yi Xu, Jiandong Ding, Jihong Guan, and Shuigeng Zhou. 2022. Epida: An easy plug-in data augmentation framework for high performance text classification. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 4742–4752. Association for Computational Linguistics.

Tianyi Zhou, Shengjie Wang, and Jeff A. Bilmes. 2021. Robust curriculum learning: from clean label detection to noisy label self-correction. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Wenxuan Zhou and Muhao Chen. 2021. Learning from noisy labels for entity-centric information extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 5381–5392. Association for Computational Linguistics.

Appendices

A More Details about Experiments

A.1 More Details about Text Classification

We use the codebase and experimental settings from EPiDA[5] (Zhao et al., 2022) to conduct our experiments. Table 6 shows the essential hyper-parameters that are used for each dataset. During the training, we first train a few epochs on the original dataset, and then finetune on the union of augmented data and original data.

For EPiDA (Zhao et al., 2022), we follow the setting in the original paper to first produce $k = 50$ augmented examples per original example using EDA, and then select top 3 scored by its Relative Entropy Maximization (REM) and Conditional Entropy Minimization (CEM) filter. The trade-off parameter between REM and CEM is set as 0.5, as in the original paper.

For Glitter (Kamalloo et al., 2022) and large-loss, similar with EPiDA, we sample 50 augmented examples first, and select the top 3 examples with the largest/smallest loss in the current run. For Re-weight (Yi et al., 2021), we use the following re-weighting equation to re-weight the augmented data in a batch:

$$w_{x_i} = \frac{\exp\left(\frac{1}{\lambda} l_{\text{CE}}\big(g(f(x_i)), y_i\big)\right)}{\sum_{x_j \in \mathcal{B}} \exp\left(\frac{1}{\lambda} l_{\text{CE}}\big(g(f(x_j)), y_j\big)\right)}$$

where w_{x_i} is the re-weighting factor for the example x_i, $\mathcal{B}$ is the current batch, and λ is a temperature parameter. The re-weighting factor is basically the softmax of the loss of the current batch.

For UDA (Xie et al., 2020a), we leverage the augmented data in consistency training. In addition to the cross-entropy loss of the original data, we jointly train with the objective that minimizing the consistency loss between original data and augmented data:

$$\mathcal{L} = \sum_{i=1}^{n} \Big(l_{\text{CE}}\big(g(f(x_i)), y_i\big) + \alpha_c \sum_{j=1}^{k} \text{KL}\big(\, g(f(x_i)) \,||\, g(f(x'_{i,j}))\,\big) \Big) \tag{5}$$

where $x'_{i,j}$ is the j-th augmented example derived from x_i. α_c is the hyper-parameter to control the effect of consistency training. It's set as 10 after sufficient parameter searching.

[5]https://github.com/zhaominyiz/EPiDA

Method	TREC		Irony		AGNews	
	1%	10%	1%	10%	0.05%	0.1%
Back-Trans. (BT)	62.55	93.62	52.29	64.69	85.39	86.35
BT+OD	62.19	94.67	57.50	64.57	85.53	86.74
BT+OD+SR	**65.02**	**95.65**	**58.10**	**65.28**	**86.03**	**86.83**

Table 5: Experiments on using back-translation as the backbone data augementation method.

Besides using EDA as the backbone data augmentation method, we also test our ODDA framework on back-translation[6] in Tab. 5. We can find that the ODDA framework can also work on back-translation, indicating a good generalizability of our framework.

A.2 More Details about Question Answering

For question answering tasks, following previous works (Sakaguchi et al., 2020; Yang et al., 2020), we use RoBERTa as the base encoder. For each question-option pair, the input format is then `[CLS] context [SEP] option [SEP]`. We take the embedding of the `[CLS]` token as the representation of the question-option pair. Then an MLP + softmax layer is put after the embeddings of the c options, and the model is optimized with cross-entropy loss given a correct option.

WinoGrande is a commonsense reasoning benchmark to explore hard coreference resolutions problems such as "The fist ate the worm, ___ was tasty" (choose from "fish" and "worm"). It's hard as it requires commonsense knowledge that "the subject of *eat* tends to be hungry and the object of *eat* tend to be tasty", while machine learning models may associate "fish" with "tasty" with larger likelihood as they frequently co-occur in human corpora. The WinoGrande dataset is composed of 5 subsets with different sizes, XS ($n = 160$), S ($n = 640$), M ($n = 2558$), L ($n = 10234$), and XL ($n = 40398$).

CommonsenseQA is a commonsense question answering dataset constructed from the commonsense knowledge in ConceptNet (Speer et al., 2017). It aims to study the commonsense relations among daily entities within certain context. For example, the correct answer to "Where would you store a pillow case that is not in use?" is "drawer". There are some distractor options such as "bedroom", which

[6]We use the implementation from the nlpaug package (https://github.com/makcedward/nlpaug)

	TREC		Irony		AGNews		Sentiment		Offense	
	1%	10%	1%	10%	0.05%	0.1%	1%	10%	0.1%	1%
Optimizer	AdamW									
Weight Decay	1e-3									
Adam ϵ	1e-8									
LR	2e-5									
Batch Size	32									
Max Length	512									
Organic Epoch	40	30	100	20	30	30	30	10	30	30
Augmentation Epoch	40	30	100	30	30	30	30	10	30	30
Evaluation Interval	1	5	1	1	5	5	5	20	1	5
Temperature τ	1	1	1	1	2	2	0.5	0.5	1	1
SR α	10	10	10	10	10	10	10	10	10	10

Table 6: Hyperparameters for text classification experiments.

is a common place where a pillow locates without the context "not in use".

The augmentation method that we use for solving commonsense question answering is Generative Data Augmentation (Yang et al., 2020). It uses three generation models to generate questions, correct answers, and distractors, respectively. Then in the data selection phase, influence function and a specifically designed heuristics that favors diverse synthetic data are used to select high-quality synthetic data. Then the model is trained with a two-stage finetuning, where they first finetune the QA model on the synthetic data, and then finetune on the original data. We use the released augmented data from Yang et al. (2020). The number of augmented instances for each dataset is presented in Table 7. The hyperparameters that are used for the experiments for QA are presented in Table 8.

B Self-Regularization

We explain the reasons why Self-Regularization can serve as a denoising channel and yield better performance. It is shown that the following finetuning method can enhance the robustness of representation learning, which provide guarantees for stochastic gradient descent algorithms by bounding some divergence between model at step t and $t+1$ (Pascanu and Bengio, 2014):

$$\begin{aligned} &\arg\min_{\Delta\theta} \mathcal{L}(\theta+\Delta\theta) \\ &\qquad s.t.\ KL(f(\cdot,\theta_f)||f(\cdot,\theta_f+\Delta\theta_f)) = \epsilon \end{aligned} \tag{6}$$

Here, f is a function that outputs vector representations, θ is the trainable parameters. An approximation to this computationally intractable equation is proposed as follows (Aghajanyan et al., 2021):

$$\begin{aligned} &\mathcal{L}(f,g,\theta) = \mathcal{L}(\theta) + \lambda KL_S(g\cdot f(x)||g\cdot f(x+z)) \\ &\qquad s.t.\ z\sim\mathcal{N}(0,\sigma^2 I) \text{ or } z\sim\mathcal{U}(-\sigma,\sigma) \end{aligned} \tag{7}$$

Here g is a function that converts the output embedding of f to a probability distribution. KL_S is the symmetric KL divergence, and z is sampled from the corresponding distribution as small perturbations. Instead of providing small perturbations using a random noise, Self-Regularization provide such perturbation with two different dropouts, which has shown to be effective in previous works (Liang et al., 2021).

Moreover, there are other empirical findings that favors the effect of self-regularization in terms of denoising. Noisy examples tend to be frequently forgotten after training for a long time (Toneva et al., 2019), since the noise conflict with what have been learned in the model and the prediction can vary. Self-regularization can be an alternative objective that mitigate the importance of the example.

	WinoGrande					CSQA
	XS	S	M	L	XL	
# Original	160	640	2,558	10,234	40,398	9,727
# Synthetic	52,346	97,733	127,509	132,849	136,052	50,014

Table 7: Number of training instances for WinoGrande and CommonsenseQA.

	WinoGrande					CSQA
	XS	S	M	L	XL	
Optimizer			AdamW			AdamW
Weight Decay			0.01			0.01
Adam ϵ			1e-6			1e-6
LR synthetic			5e-6			5e-6
LR organic			1e-5			1e-5
Batch Size			16			16
Max Length			70			70
Synthetic Epoch	1	1	1	1	1	1
Organic Epoch	10	8	5	5	5	5
LR Decay			Linear			Linear
Warmup Ratio			0.06			0.06
SR Warmup Steps	2000	5000	5000	7000	7000	2500
τ	2	1	1	1	1	1
α	0.5	0.1	1.0	0.5	0.5	0.5

Table 8: Essential Hyperparameters for WinoGrande and CommonsenseQA.