**ZEROGEN**: Self-Guided High-Quality Data Generation in Efficient Zero-Shot Learning

Jiahui Gao, Renjie Pi, Yong Lin, Hang Xu, Jiacheng Ye, Zhiyong Wu, Xiaodan Liang, Zhenguo Li, Lingpeng Kong

1The University of Hong Kong 2Hong Kong University of Science and Technology 3Huawei Noah’s Ark Lab 4Shanghai AI Lab 5Sun Yat-sen University

{sumiler,lpk}@hku.hku, {rpi,ylindf}@connect.ust.hk, {xu.hang,li.zhenguo}@huawei.com, yejc19@fudan.edu.cn, wuzhiyong@pjlab.org.cn, xdliang328@gmail.com

Abstract

Nowadays, owing to the superior capacity of the large pre-trained language models (PLM), the PLM-based zero-shot learning has shown promising performances on various natural language processing tasks. There are emerging interests in further exploring the zero-shot learning potential of PLMs. Among them, ZeroGen (Ye et al., 2022a) attempts to purely use PLM to generate data and train a tiny model without relying on any task-specific annotation. Despite its remarkable results, we observe that the synthesized data from PLM contains a significant portion of samples with low quality, overfitting on such data greatly hampers the performance of the trained model and makes it unreliable for deployment. Since no gold data is accessible in zero-shot scenario, it is hard to perform model/data selection to prevent overfitting to the low-quality data. To address this problem, we propose a noise-robust bi-level re-weighting framework which is able to learn the per-sample weights measuring the data quality without requiring any gold data. With the learnt weights, clean subsets of different sizes can then be sampled to train the task model. We theoretically and empirically verify our method is able to construct synthetic dataset with good quality. Our method yields a 7.1% relative improvement than ZEROGEN on average accuracy across five different established text classification tasks.

1 Introduction

Owing to the superior generative capacity of large-scale pre-trained language models (PLMs), there has been an emerging trend of using such powerful models to generate training data. Numerous attempts have been made towards using the generated data to augment the training set (Anaby-Tavor et al., 2020; Puri et al., 2020; Kumar et al., 2020) or generate informative samples for humans to label (Liu et al., 2022), which have achieved promising results. Nevertheless, previous work still highly depends on in-domain gold data or human labeling, which can be expensive or unrealistic.

Recently, a new line of research pushes the envelop further (Ye et al., 2022b; Meng et al., 2022): they purely rely on the PLM-generated data for different downstream tasks. Specifically, Ye et al. (2022b) use PLM to generate synthetic training data and train a tiny model from scratch with only such generated data. This approach demonstrates better performance than the traditional prompt-based zero-shot PLM counterparts while being much more efficient during inference.

In the above paradigm, the quality of the generated dataset becomes crucial for the model’s performance. Unfortunately, we observe across many downstream tasks that the actual test performance starts declining rapidly after a few training epochs (e.g. IMDb in Figure 1), demonstrating a clear sign of overfitting to erroneous data (Arpit et al., 2017). This problem becomes more detrimental in zero-shot scenario: since no clean validation set is available, the model cannot be validated during training and thus overfits to the low-quality generated data.

Figure 1: Training and testing accuracy of Bi-LSTM model trained on synthetic datasets. After training for more epochs, the testing performance starts to deteriorate significantly, showing that the model start to learn and fit the erroneous data.
available, it is hard to check the trained model’s real performance on the task. A model that achieves small training loss may overfit to the low-quality data and leads to severe performance degradation.

Numerous attempts have been made towards tackling model training with noisy dataset. One line of research focuses on resampling or reweighting approaches, which requires either heuristically setting a threshold for loss values to exclude noisy data (Han et al., 2018; Jiang et al., 2018; Yu et al., 2019), or a clean validation set to learn the loss reweighting (Ren et al., 2018; Shu et al., 2019). These constraints make them impractical in our zero-shot setting. Another direction explores a family of noise-robust loss functions (Ghosh et al., 2017; Wang et al., 2019) with theoretic guarantees to train the model using noisy data only. However, these losses are typically harder to optimize for DNNs and may hurt the performance (Zhang and Sabuncu, 2018; Wang et al., 2019).

We draw inspiration from previous noise-robust learning methods and put forth an idea: can we adopt noise-robust losses \( l_{\text{robust}} \) to optimize the per-sample weights (rather than the model) to improve the quality of the dataset? This counteracts the shortcomings of both the above mentioned paradigms: (1) we do not need gold data to learn the per-sample weights or manually setting the threshold due to the use of \( l_{\text{robust}} \); and (2) we use \( l_{\text{robust}} \) to optimize the per-sample weights rather than the model parameters, which prevents affecting the model’s performance. The learnt weights can then be leveraged to sample clean subsets to train the model following standard empirical risk minimization (ERM) training scheme.

To achieve this, we formulate the learning of sample reweighting into a bilevel optimization problem: in the inner loop, we train the task model with the weighted training loss; in the outer loop, the noise-robust loss is adopted to guide the learning of the sample weights. Note that both the inner and outer objectives are calculated on the synthetic training set. The two procedures are performed alternatively, which eventually generates a set of weights, such that the erroneous samples are associated with small weights and those informative data are associated with large weights. We theoretically verify that the gradient of the outer objective \( l_{\text{robust}} \) with respect to the per-sample weights stays the same, regardless of whether the validation set is noisy or clean.

Our contribution in this paper are as follows:

- Our paper proposes an end-to-end sample-reweighting framework to improve the quality of the synthetic training dataset generated by PLM, without the aid of any labeled in-domain data.
- We theoretically justify that our method is able to recover a clean dataset reliably.
- We conduct extensive experiments and show our method yields a 7.1% relative improvement than ZEROGEN on average accuracy.

2 Background

2.1 Prompt-based Zero-Shot Learning

We first introduce prompt-based zero-shot prediction (named PROMPTING). The idea is to use large-scale PLM to infer the correct label based on the query example and the task-specific prompt. For example, in text classification task \( D = (X, Y) \) (e.g. IMDb), given a manually-designed prompt \( T(\cdot) \) and a query example \( x_i \in X \), PROMPTING constructs a sentence \( T(x_i) \) (e.g. “The movie review in <MASK> sentiment is: <x_1>”). The PLM \( P \) is expected to model the probability of class \( y_i \in Y \) for \( x_i \) as

\[
p(y_i|x_i) = P(M(y_i)|T(x_i)),
\]

where \( M(\cdot) \) is a verbalizer that maps each label/class \( y_i \) to a label word/words \( v_i \) in \( P \)’s vocabulary \( V \). For example, "positive" is a label word \( v_i \) representing the positive class \( y_i = 1 \) and is placed on the \(<MASK>\) position in \( T(\cdot) \). During the whole process, the PLM is frozen and directly do inference without any training.

PROMPTING has achieved remarkable success owing to the linguistic and factual knowledge encoded within its parameters (Jawahar et al., 2019; Petroni et al., 2019; Jiang et al., 2020b). However, PROMPTING does not fully extract the useful knowledge from PLMs and still needs to conduct inference on a cumbersome PLM. Therefore, a new line of research (Ye et al., 2022b) endeavor to make zero-shot learning more practical and efficient by using the data generated by PLM to train a tiny model from scratch.

2.2 Efficient Zero-Shot Learning via Data Generation

The generative efficient zero-shot learning paradigm is proposed by ZEROGEN (Ye et al., 2022b).

...
2022b). Given a downstream task, the paradigm first generates task-specific synthetic data with the help of large-scale PLM and task-related prompts. Then, based on the synthetic data, the paradigm trains a tiny task model (TAM) for different tasks. During the whole process, no human annotations are required and the TAM is much more efficient than its PLM counterparts.

**Synthetic Data Generation.** The work first generates a synthetic dataset $D_{\text{syn}} = (\mathcal{X}_{\text{syn}}, \mathcal{Y}_{\text{syn}})$ using a left-to-right PLM $\mathcal{P}$. The idea is to use model $\mathcal{P}$ to generate the input $x_{\text{syn}}$ based on a pseudo label $y_{\text{syn}}$. For example, in text classification task, a class label $y_{\text{syn}}$ is uniformly sampled:

$$y_{\text{syn}} \sim U(y_1, y_2, \ldots, y_c), \quad (2)$$

where $c$ is the number of classes. The pseudo label $y_{\text{syn}}$ is transformed into a label-descriptive prompt $\mathcal{T}(\mathcal{M}(y_{\text{syn}}))$ to generate $x_{\text{syn}}$:

$$x_{\text{syn}} \sim \mathcal{P}(\cdot|\mathcal{T}(\mathcal{M}(y_{\text{syn}}))). \quad (3)$$

The generated $x_{\text{syn}}$ and pseudo label $y_{\text{syn}}$ can be paired to construct a pseudo training dataset $D_{\text{syn}}$. During the generation of $(\mathcal{X}_{\text{syn}}, \mathcal{Y}_{\text{syn}})$, nucleus sampling (Holtzman et al., 2020)) is adopted as the decoding strategy to promote more diversity in the generated dataset.

**Efficient Training and Inference.** For the target to efficiently training and inference, ZEROGEN then trains TAM (e.g., 1-layer Bi-LSTM) on the synthetic dataset $D_{\text{syn}}$ for different downstream tasks.

### 3 ZEROGEN$^+$

Although in efficient zero-shotting scenario, ZEROGEN have shown that it can successfully train a TAM from scratch with the data generated by PLMs, we empirically observe that the actual test performance rapidly declines after several training epochs, which indicates overfitting to erroneous data (shown in Section 4.2). This phenomenon promotes us raise the following question: how can we identify and remove the erroneous data without relying on any gold samples?

To tackle the above issue, we propose a framework for clean data construction, named ZEROGEN$^+$ (Figure 2). More specifically, we propose a robust sample reweighting framework via bilevel optimization to learn an weight measuring the data quality for each data without supervision from any gold data. The learnt weights are then used to sample a clean subset then be used to train the TAM under standard empirical risk minimization (ERM) procedure. We elaborate our proposed method in the following sections.

#### 3.1 Framework of ZEROGEN$^+$

**Noise-robust Loss Functions.** There is a family of loss functions that possess the “symmetric” property:

$$\sum_{i=1}^{c} \ell_{\text{robust}}(f(x), y) = Q, \forall f, x, \quad (4)$$

![Figure 2: The framework of ZEROGEN$^+$. Our noise-robust bi-level framework aims to learn per-sample weight measuring the data quality without requiring any gold data supervision. More specifically, in the inner loop, we train the tiny model with the weighted training loss; in the outer loop, we adopt a noise-robust loss to guide the learning of the sample weights. Note that both the inner and outer objectives are calculated on the same synthetic training set. ‘Syn’ is short for synthetic.](image-url)
where $Q$ is a constant. Previous work (Wang et al., 2019; Zhang and Sabuncu, 2018; Ghosh et al., 2017) has shown that these loss functions are robust to uniform label noise when the majority of the training samples are correctly labelled, suggesting that the global minimizer of $l_{\text{robust}}$ is noise-tolerant. However, those approaches mainly adopt the symmetric losses to optimize the model parameters directly, and Wang et al. (2019) has shown that these losses are difficult to optimize for neural networks, which needs careful tuning and may sacrifice the trained model’s performance. This motivates us to raise a question: can we use the symmetric losses as objective functions to select data?

Driven by the above intuition, we propose to leverage $l_{\text{robust}}$ to “optimize” our training data, rather than the model parameters. Specifically, we solve the task via an end-to-end sample reweighting framework, which adopts $l_{\text{robust}}$ as the objective to optimize sample weights $w$. The learnt $w$ can then indicate the quality of each sample. Since we optimizing the sample weights $w$ rather than the model parameters, we do not risk sacrificing the model’s performance.

In our implementation, we adopt the reversed cross entropy loss ($l_{\text{rce}}$), which has the following form:

$$l_{\text{rce}}(\hat{y}_i, y_i) = -\sum_{k=1}^{K} \hat{y}_i^k \log(y_i^k)$$

where $\hat{y}_i$ is the predicted class of TAM and $K$ is the number of classes; in the case of $y_i^k = 0$, $\log(0)$ is approximated to a constant $A$. In essence, compared with the ordinary cross entropy loss ($l_{\text{ce}}$), the reversed cross entropy loss ($l_{\text{rce}}$) swaps the order of network prediction $\hat{y}$ and label $y$. More analysis of the reversed cross entropy loss are presented in (Wang et al., 2019).

**Bilevel Framework for Robust Sample Reweighting.** Our proposed noise-robust sample reweighting framework (as shown in Figure 2) can be outlined as follows:

$$w^* \in \arg \min_{w} \mathcal{L}_{\text{rce}}(\theta^*(w); \mathcal{D}_{\text{syn}}^w), \quad (5)$$

$$\text{s.t. } \theta^*(w) \in \arg \min_{\theta} \mathcal{L}_{\text{ce}}(\theta; \mathcal{D}_{\text{syn}}^t(w))$$

where

$$\mathcal{L}_{\text{rce}}(\theta^*(w); \mathcal{D}_{\text{syn}}^w) = \mathbb{E}[l_{\text{rce}}(f(x; \theta^*(w)), y)],$$

$$\mathcal{L}_{\text{ce}}(\theta; \mathcal{D}_{\text{syn}}^t(w)) = \mathbb{E}[w(x, y) l_{\text{ce}}(f(x; \theta), y)],$$

$\theta$ is the parameters of the TAM; $\theta^*$ is the parameters trained to convergence; $w \in \{W : \mathcal{X} \times \mathcal{Y} \rightarrow [0, 1]\}$ is a re-weighting function that assign a weight to each sample based on $x$ and $y$, which indicates the importance of each training sample. Notably, this also includes the case of assigning independent weight to each sample.

Intuitively, the above formulation aims to learn per-sample re-weighting $w$, such that the task model trained with the reweighted loss achieves low loss value averaged on the entire dataset in terms of the outer robust loss. More specifically, in the inner loop, we train the task model with the weighted training loss; in the outer loop, we adopt a noise-robust loss to guide the learning of the sample weights. Note that both the inner and outer objectives are calculated on the same training set $\mathcal{D}_{\text{syn}}$. By alternatively optimizing the inner and outer objectives, the sample weights associated with erroneous data can gradually become small and those associated with clean data can become larger, thus creating the meaningful separation for subset sampling.

**Truncated Back-propagation for Meta Gradient.** For solving the above bilevel optimization problem, the gradient of $w$ (meta gradient) can be calculated as following:

$$\nabla_w \mathcal{L}_{\text{rce}} = \nabla \theta \mathcal{L}_{\text{rce}}|_{\theta^*} \nabla_w \theta^* \approx \nabla \theta \mathcal{L}_{\text{rce}}|_{\theta^*} \nabla_w \theta_T \quad (6)$$

$$= \nabla \theta \mathcal{L}_{\text{rce}}|_{\theta_T} \sum_{i \leq T} \left[ \prod_{k < j} \frac{\partial^2 \mathcal{L}_{\text{rce}}}{\partial \theta \partial \theta^T} |_{\theta_T = k-1} \right] \frac{\partial^2 \mathcal{L}_{\text{rce}}}{\partial \theta \partial \theta^T} |_{\theta_T = j-1} \approx \nabla \theta \mathcal{L}_{\text{rce}}|_{\theta_T} \frac{\partial^2 \mathcal{L}_{\text{rce}}}{\partial \theta \partial \theta^T} |_{\theta_T = j-1}, \quad (7)$$

where Eqn. (6) follows chain rule, Eqn. (7) approximates $\theta^*$ by $\theta_T$ obtained from $T$ steps of inner loop gradient descent and Eqn. (8) performs 1-step truncated backpropagation (Shaban et al., 2019). Note that $\nabla_w$ is also referred to as the meta gradient in the following sections.

**Clean Subset Sampling.** Our reweighting framework enables us to derive a set of continuous reweights $w$, which encodes the importance of each data and can well separate the noisy samples from those clean ones, as shown in Figure 3. With those reweights, we can sample a clean subset with arbitrary budgets from it and train the task models via unweighted ERM. More specifically, we let the indicator variable $I_i$ of data point $\mathcal{D}_{\text{syn}}^t$ be a Bernoulli random variable:
Algorithm 1 Bilevel Robust Sample Reweighting

Require: a TAM with parameters \( \theta \), generated dataset \( D_{\text{syn}} \) and subset size \( K \), outer step size \( \eta \), outer iterations \( T \).
1: Initialize sample weightings \( w \) to be 0.5.
2: for training iteration \( t = 1, 2 \ldots T \) do
3: Conduct weighted training on TAM and obtain \( \hat{\theta}^* \)
4: \( D_{\text{syn}}^v \leftarrow \) Uniformly sample from \( D_{\text{syn}} \)
5: Calculate meta gradient \( \nabla_{\theta} \mathcal{L} \) via Eqn. (8).
6: Update \( w \) using \( \nabla_{w} \mathcal{L} \) by gradient descent
   \( w^{t+1} \leftarrow w^t - \eta \nabla_{w} \mathcal{L} \)
7: end for

Output: Optimized sample weights \( w^* \).

The whole algorithm is described in Algorithm 1. Our framework enjoys the following advantages against the existing noise-robust learning methods:

- Compared with the methods that select samples with small loss values (Han et al., 2018; Jiang et al., 2018), our framework is end to end and does not require manually setting the loss threshold.
- Since both the inner and outer objectives are calculated on the same training set, we do not need the supervision from any gold data, which is required by previous meta-learning based methods (Ren et al., 2018; Shu et al., 2019).
- Compared with methods that train the model with robust losses (Zhang and Sabuncu, 2018; Wang et al., 2019), our approach leverages the robust loss to learn the sample-weights, which is easier to optimize and does not hurt the model’s performance.

3.2 Theoretical Analysis

Though we don’t have clean data as validation set, our method still enjoys some favorable theoretical properties. The following theorem shows that the meta gradient guided by the noisy validation set is actually in the same direction of that guided by clean validation set by assuming infinite samples.

**Theorem 1** (Infinite Sample Gradient). Assuming infinite validation samples generated with uniform noise rate \( \eta < \frac{1}{2c} \), given any symmetric loss function \( \ell \), we have

\[
\frac{\partial \mathcal{L}(\theta^*(w), D_{\text{syn}}^v)}{\partial w} = \frac{c - 1 - c\eta}{c - 1} \frac{\partial \mathcal{L}(\theta^*(w), D_{\text{clean}}^v)}{\partial w},
\]

where \( \mathcal{L}(\theta^*(w), D) = E_{(x,y) \sim D} \ell(f(x; \theta^*(w)), y) \).

Refer to Appendix A for full proof. Note that uniform noise is a common assumption in existing literature (Ghosh et al., 2017; Wang et al., 2019). (Wang et al., 2019) also verifies the rev-cross-entropy is symmetric. Theorem 1 ensures that the meta gradient direction generated by the noisy validation set is in the same direction with that generated by a clean set. Thus our method is equivalent to finding a sample weight that maximizes the performance on a clean validation set.

**Theorem 2** (Finite Sample Generalization). Suppose we have access to synthetic datasets \( D_{\text{syn}}^v \) and \( D_{\text{syn}}^v \) both with finite samples \( N \). Let \( \hat{\theta}^*(w) \) be the deterministic mapping from \( w \) to \( \theta \) defined in the inner of Eqn.5 given \( D_{\text{syn}}^v \). Assuming the output of loss function \( \ell_{\text{rec}} \) is upper bounded by \( M \), the carnality of \( \mathcal{V} \) is \( |\mathcal{V}| \), the outer loop of Eqn.(5) is solved within \( \epsilon \)-approximately such that we have a \( w \) satisfying the following condition:

\[
\mathcal{L}_{\text{rec}}(\hat{\theta}^*(w); D_{\text{syn}}^v) \leq \arg \min_w \mathcal{L}_{\text{rec}}(\hat{\theta}(w); D_{\text{syn}}^v) + \epsilon,
\]

we then have with probability at least \( 1 - \delta \),

\[
\mathcal{L}_{\text{rec}}(\hat{\theta}^*(w)) \leq \arg \min_w \mathcal{L}_{\text{rec}}(\theta^*(w); D_{\text{syn}}^v)
\]

\[+ \epsilon + \displaystyle \sqrt{\frac{2 \ln(|\mathcal{V}|/\delta)}{N}}.
\]

Refer to Appendix B for the full proof. Theorem 2 characterizes the generalization ability of Eqn.(5). If we obtain a approximated solution of the bilevel problem Eqn.(5), this solution will approach the oracle \( w \) and the corresponding \( \theta \) given sufficient large number of samples. Theorem 1 already shows that our method will find the solution guided by a clean dataset if the samples are infinite. Theorem 2 indicates when the samples are finite, our solution will converge to the finite sample solution by \( O(1/\sqrt{N}) \). These two theorems together show the theoretical soundness of our methods.
### 4 Experiments

#### 4.1 Setup

**Datasets** We evaluate our method on five different natural language classification tasks, including IMDb (Maas et al., 2011), SST-2 (Socher et al., 2013), Rotten Tomatoes (Pang and Lee, 2005), Elec (McAuley and Leskovec, 2013) and Yelp (Zhang et al., 2015). IMDb, SST-2, and Rotten Tomatoes are sentiment classification benchmarks containing positive/negative movie reviews. Elec and Yelp are binary classification tasks consisting of electronic product reviews and restaurant reviews. We choose electronics and restaurant reviews as it seemed to be very different from movie reviews. Sample sizes of datasets are described in Table 1. We use accuracy as the evaluation metric.

**Baselines** We compare our proposed method with the following baselines:

- **PROMPTING.** The prompt-based zero-shot classification method based via PLMs (Brown et al., 2020; Gao et al., 2021).

- **ZEROGEN.** A recent zero-shot learning work via dataset generation (Ye et al., 2022a). By designing task-specific prompts, large PLMs are used to generate a synthetic dataset and then light-weight classifiers are trained on the generated dataset.

**Implementation Details** We compare the baselines using GPT2-XL (Radford et al., 2019) as PLM. For text generation, we use Nucleus Sampling (Holtzman et al., 2020) with $p = 0.9$ as the decoding strategy and use GPT2-XL as the generator. For fair comparison, we use the best prompts designed by (Ye et al., 2022a) for data generation. For task model training, we use 1-layer Bi-LSTM and DistilBERT-base as the light-weight classifiers. The bilevel procedure is iterated for 50 times for each task. For more details (e.g. full prompts, training details), please refer to Appendix C.

#### 4.2 Empirical observation of noisy data

From Figure 1, we observe that the actual test performance starts declining rapidly after a few epochs of training, which is an implication that the model starts to overfit on erroneous samples. However, if we use data constructed by ZEROGEN framework, the problem can be solved. With longer training epochs, the data of ZEROGEN consistently helps improving the model’s actual performance, proving that ZEROGEN can generate clean synthetic data without noise.

#### 4.3 Main Experiment

We present our main experiment results in Table 1. Specifically, we compare the performance of our ZEROGEN with baselines methods and datasets mentioned in 4.1. In addition, we also compare with the results obtained by supervised training with gold data. For fair comparison, the prompts used for generation is the same with ZEROGEN (Ye et al., 2022b). Experiments are conducted on both LSTM and DistilBERT for comparison. We observe that our ZEROGEN achieves considerable performance gain over the ZEROGEN baseline across all the tasks. Interestingly, the improvement is much more prominent for LSTM, which beats ZeroGen by a margin of 7.1% on average across all tasks. We conjecture the reason is that the pretrained models are inherently more robust to noisy training data, which is also pointed out in (Hendrycks et al., 2019). Surprisingly, on IMDb, Rotten Tomatoes and Yelp, ZEROGEN-LSTM even outperforms ZEROGEN-DistilBERT while

| TAM | #Param | Setting | IMDb | SST-2 | Rotten Tomato | Elec | Yelp | Avg |
|-----|--------|---------|------|-------|---------------|------|------|-----|
| DistilBERT | 66M | SUPERVISED | 87.24 | 89.68 | 83.67 | 92.63 | 95.42 | 89.73 |
| LSTM | ∼7M | PROMPTING | 84.60 | 76.30 | 77.49 | 86.36 | 91.30 | 83.21 |
| - | 1.5B | | | | | | | |
| DistilBERT | 66M | ZEROGEN | 84.28 | 87.27 | 83.02 | 87.19 | 87.58 | 85.87 |
| | | ZEROGEN+ | 89.38 | 89.45 | 84.52 | 89.01 | 89.19 | 88.31 |
| LSTM | ∼7M | ZEROGEN | 79.80 | 78.40 | 73.45 | 80.48 | 84.95 | 79.42 |
| | | ZEROGEN+ | 84.10 | 84.58 | 83.21 | 84.22 | 89.06 | 85.03 |

Table 1: Evaluation results for ZEROGEN framework on two different scales of TAM. The scale of synthetic dataset is 200k for both ZEROGEN and ZEROGEN+. ZEROGEN results are collected from Ye et al. (2022b).
which uses gold data for calculating the outer objective.

Table 2: Evaluation results for ZERO

| Method  | Outer | IMDb | Elec | Yelp | Rotten |
|---------|------|------|------|------|-------|
| SUPERVISED | -    | 84.60 | 86.36 | 91.30 | 77.49 |
| ZEROGEN  | -    | 71.52 | 80.48 | 84.95 | 73.45 |
| SmartGen  | Gold | 82.34 | 84.71 | 88.83 | 80.05 |
|          | Syn. | 84.10 | 84.22 | 88.59 | 83.21 |

Table 2: Evaluation results for ZERO framework using different outer objectives. ‘Gold’ represents using task-specific labeled data (the standard training set) as data used in outer loop. ‘Syn.’ represents using synthetic data in the outer loop.

4.4 Ablation Study and Analysis

Table 3: Evaluation results of ZERO+/LSTM on different data sizes. For the subsets, models are directly trained on the selected set. For the 1,000k results of ZERO+, we reported the model trained on 1,000k synthetic data using weighted $L_{ce}$. ZERO and ZERO+ are short for ZEROGEN and ZEROGEN+, respectively.

| Size  | IMDb | Elec | Yelp |
|-------|------|------|------|
| 1,000k | 78.29 | 86.56 | 82.68 | 84.63 | 86.28 | 90.38 |
| 10k   | 62.40 | 72.05 | 74.46 | 75.84 | 75.22 | 80.67 |
| 20k   | 65.12 | 79.96 | 75.78 | 77.52 | 79.55 | 82.88 |
| 50k   | 68.12 | 81.14 | 78.14 | 81.97 | 80.81 | 85.82 |
| 100k  | 71.28 | 82.09 | 80.25 | 83.68 | 82.97 | 88.41 |
| 200k  | 71.52 | 84.10 | 80.48 | 84.22 | 84.95 | 89.06 |

Table 3: Evaluation results of ZEROGEN+-LSTM on different data sizes. For the subsets, models are directly trained on the selected set. For the 1,000k results of ZEROGEN+, we reported the model trained on 1,000k synthetic data using weighted $L_{ce}$. ZERO and ZERO+ are short for ZEROGEN and ZEROGEN+, respectively.

having much fewer parameters (7M vs 66M).

4.4 Ablation Study and Analysis

Synthetic Data vs Gold Data Supervision. Note that a key advantage of our proposed framework is that we do not require gold data to optimize the sample weights. Here, we compare our ZEROGEN+ to the bilevel reweighting method which uses gold data for calculating the outer objective in Table 2. Specifically, our ZEROGEN+ calculates the robust outer objective on the noisy synthetic data, while the counterpart calculates the outer objective on the gold data. From Table 2, we can observe that our method can achieve similar performances with using gold data in the outer iteration, which verifies that our robust outer objective on the noise validation set can equivalently supervise the optimization of sample weights as the clean validation set does.

Prediction Ability of Different Data Sizes. To further investigate the prediction ability of synthetic data of ZEROGEN+, we compare model results trained in different data sizes. More specifically, we sample datasets with small amount of samples from 1,000k original synthetic data. From the result in Table 3, we find that model results of ZEROGEN+ not only surpass ZEROGEN in the same data size, but also can achieve much better performance than the model trained in a larger data size. For example, in IMDb, the model fitted on 20k data from ZEROGEN+ even achieves better performance than that trained on 1,000k data from ZEROGEN. The results show the superiority of data quality constructed by ZEROGEN+.

Diversity and Correctness. We measure the diversity and correctness of the generated datasets following Ye et al. (2022b). Self-BLEU4 is used to measure the diversity, and the RoBERTa-Large model finetuned on in-domain labeled data is used as the oracle model to measure the correctness. For ZEROGEN and ZEROGEN+, 10,000 samples are randomly generated for evaluation. The result in Table 4 shows that our selected clean dataset is more diverse than data generated by ZEROGEN. One interesting thing to note is that in Elec and Yelp, average correctness of ZEROGEN is slightly lower than ZEROGEN. In addition, the samples associated with smallest weights have much lower correctness than average, while the ones with largest weights also have slightly lower correctness. This is expected as the data with highest correctness may be redundant or too simple, while the challenging and informative samples are often harder to classify and thus have lower correctness. The results further verifies that ZEROGEN+ effectively separates the informative samples with the ones that are either redundant or erroneous, which can not be done with the heuristic methods that manually sets a threshold of loss values to separate clean and noisy data (Han et al., 2018; Jiang et al., 2018; Yu et al., 2019), which may keep the simple redundant samples and remove the hard informative ones.

| Method  | IMDb | Elec | Yelp |
|---------|------|------|------|
| Diversity |       |      |      |
| Gold    | 0.30 | 0.29 | 0.29 |
| ZEROGEN | 0.15 | 0.12 | 0.14 |
| ZEROGEN+ | 0.14 | 0.10 | 0.11 |

| Correctness(%) |       |      |      |
| Gold           | 96.22 | 96.60 | 98.35 |
| ZEROGEN        | 75.86 | 93.58 | 94.47 |
| ZEROGEN+       | 82.27 | 88.87 | 90.78 |
| “Top samples”  | 86.00 | 84.20 | 83.33 |
| “Bottom samples” | 4.57 | 61.50 | 44.66 |

Table 4: Diversity and Correctness evaluation for different methods. “Gold” refers to the standard dataset with human annotations. “Top samples” and “Bottom samples” represent 10,000 samples with highest weights and lowest weights respectively.
Figure 3: Histogram of learned weights in IMDb synthetic dataset (1,000k) using Bi-LSTM as task model. The initial weights are 0.5 for all samples. With updating more iterations, the weight’s distribution interval (in x-axis) is gradually changing from a small range to a large range, which indicates our method is able to clearly differentiates high-quality data and erroneous data by large/small weights.

| Text <X>                                                                 | Label <Y> | Noisy Type |
|--------------------------------------------------------------------------|-----------|------------|
| The film does a great job of capturing the fear and battle that so many U.S. troops have experienced during the seven-year war in Afghanistan. | Neg.      | Noisy Y    |
| This long, pompous, chain-smoking movie makes a big hit out of not very much at all. One of the worst cult films ever made. 2D CGI animations of zombies and comic-book characters. Some bad acting, technical problems, cheap gimmicks and script that is | Pos.      | Noisy Y    |
| Helping kids to accept all kids as distinct, independent individuals.     | Neg.      | Unrelated X|
| There is a huge dilemma in the resolution of the movie that deals with powerful issues concerning the breakdown of male dominance in the family, relationships, or society. | Neg.      | Unrelated X|
| Despite its oddball structure and topsy-turvy interactions between characters, this surprisingly zany animated film succeeds where so many animated films fail. Not worth the time for the main actors, but for the almost movie has a very good story that puts many sci-fi movies of the past to shame. Wonder Woman is a big-budget superhero blockbuster that turns the spotlight on the potential of a woman leader… but the movie is ultimately unfulfilling and laden with female stereotypes. | Pos.      | No Noise   |
| Pos.                                                                     | No Noise   |
| Neg.                                                                     | No Noise   |

Table 5: Examples of removed data(low weights) and selected data(high weights) in IMDb synthetic dataset.

Analysis of Removed and Selected Examples. First, we analyze the removed erroneous data. We take IMDb synthetic dataset as the example. From the observation, we find that most of the data with small weights have noise label(Noisy Y), which indicates the class of X is wrongly labeled by PLM during generation(see Table 5). Besides, there are small part of erroneous data which contain unrelated X, which indicates the generated text have no obvious emotional tendency and cannot be categorized to any class. From Figure 3(d), we find the percentage of the erroneous data is small, but it significantly degrades the model performance(from 86.56 to 78.29 shown in Table 3). Then, we find the data selected by ZEROGEN+ are actually well-written transitional complex sentences(bottom part of Table 5), which verifies that ZEROGEN+ tends to select correct and challenging samples.

5 Related Works

5.1 Zero-shot Learning with PLM
In contrast to finetuning and few-shot learning which rely on human-annotated data (gold data) to supervise model training, zero-shoting learning on PLM studies how to do prediction without access to any task-specific data. The most popular zero-shot learning method is the prompt-based zero-shot prediction(i.e. PROMPTING) proposed by GPT (Radford et al., 2019; Brown et al., 2020), which directly do prediction via PLM without any training. With well-designed prompts, large-scale PLMs have shown its notable zero-shot learning ability in various natural language processing tasks(Jiang et al., 2020a; Shin et al., 2020; Reynolds and McDonell, 2021; Mishra et al., 2021).

Recent work investigates the zero-shot learning on PLM in a generative way. Wang et al. (2021)
uses few-shot unlabeled in-domain samples as the prompts to generate data. Schick and Schütze (2021) and Meng et al. (2022) use hand-crafted task-dependent prompts to synthesize data. These studies then use the synthetic data to finetune another PLM for downstream training. To further investigate PLM’s zero-shot ability, Ye et al. (2022a) proposes ZEROGAN to study an extreme scenario, called Efficient Zero-shot Learning.

5.2 Noise Robust Learning

The previous methods tackling data noise problem can be roughly categorized into two groups: (1) heuristic approaches based on loss values that rely on the assumption that the network learns easy samples first, which adopt either resampling (Han et al., 2018; Jiang et al., 2018; Yu et al., 2019), loss reweighting (Thulasidasan et al., 2019; Konstantinov and Lampert, 2019; Ren et al., 2018; Shu et al., 2019), or label correction (Ma et al., 2018; Kremer et al., 2018; Reed et al., 2014). These works require either manually set a threshold for the loss value or a clean validation set, which makes their performance questionable in zero-shot scenario. (2) works in another line train the network with noise-robust loss (Ghosh et al., 2017; Zhang et al., 2019), or label correction (Ma et al., 2018; Kremer et al., 2018; Reed et al., 2014). These works require either manually set a threshold for the loss value or a clean validation set, which makes their performance questionable in zero-shot scenario. (2) works in another line train the network with noise-robust loss (Ghosh et al., 2017; Zhang et al., 2019), or label correction (Ma et al., 2018; Kremer et al., 2018; Reed et al., 2014). Despite they learn a robust classifier in theory, they are typically difficult to train the DNNs and result require more hyper-parameter tuning (Zhang and Sabuncu, 2018; Ma et al., 2020; Liu and Guo, 2020; Xu et al., 2019; Wang et al., 2019). To this end, we take advantages from both lines of research and design an end-to-end framework which can reliably filter out harmful data without a clean validation set.

6 Conclusion

In this paper, we investigate the noisy data problem in efficient zero-shot learning scenario. We successfully build a bi-level re-weighting framework to construct a clean synthetic dataset without relying on any task-specific labeled data. Without any human annotations, our methods achieve significant improvement than the basic efficient zero-shot method via data generation. We hope this paper can provide insights for improving data quality, and inspire more exploration in data-generation-based zero-shot learning via large-scale PLM.

A Proof for Theorem 1

Proof. (Wang et al., 2019) shows that the reversed cross entropy is robust under uniform loss. We add the proof of them for completeness. Let \( \tilde{D} \) and \( \tilde{y} \) denote the noisy dataset and label.

\[
\mathcal{L}(\theta, \tilde{D}) = \mathbb{E}_{x, \tilde{y}}[\ell_{\text{ce}}(f(x; \theta), \tilde{y})] \\
= \mathbb{E}_{x, \tilde{y}}[\ell_{\text{ce}}(f(x; \theta), \tilde{y})] \\
= \mathbb{E}_{x, \tilde{y}}[(1 - \eta)\ell_{\text{ce}}(f(x; \theta), y) \\
+ \frac{\eta}{K - 1} \sum_{j \neq y} \ell_{\text{ce}}(f(x; \theta), j)] \\
= \mathbb{E}_{x, \tilde{y}}[\frac{K - 1 - \eta}{K - 1} \ell_{\text{ce}}(f(x; \theta), y)] + \frac{\eta KC}{K - 1} \\
= \frac{K - 1 - \eta}{K - 1} \mathcal{L}(\theta, D) + \frac{\eta KC}{K - 1}
\]

Then we have

\[
\frac{\partial \mathcal{L}(\theta^*(w), D^{\text{syn}}_w)}{\partial w} = \frac{\partial \mathcal{L}(\theta^*(w), D^{\text{syn}}_w) \partial \mathcal{L}(w)}{\partial w} \\
= \frac{K - 1 - \eta}{K - 1} \mathcal{L}(\theta^*(w), D^{\text{clean}}_w) \frac{\partial \mathcal{L}(w)}{\partial w} \\
= \frac{K - 1 - \eta}{K - 1} \mathcal{L}(\theta^*(w), D^{\text{clean}}_w) \frac{\partial \mathcal{L}(w)}{\partial w}
\]

B Proof for Theorem 2

Proof. Let \( D^{\text{syn}}_w \) denotes the dataset that replaces any one element of \( D^{\text{syn}}_w \) with arbitrary \( x \), it is easy to know that

\[
|\mathcal{L}_{\text{ce}}(\mathbf{\tilde{\theta}}^*(w); \tilde{D}^{\text{syn}}_w) - \mathcal{L}_{\text{ce}}(\mathbf{\hat{\theta}}^*(w); \tilde{D}^{\text{syn}}_w)| \leq \frac{M}{N}
\]

holds for any \( w \). Then by the bounded difference inequality (Corollary 2.21 of (Wainwright, 2019)), given \( w \), we have with probability \( 1 - \delta \),

\[
\mathcal{L}_{\text{ce}}(\mathbf{\hat{\theta}}^*(w); \tilde{D}^{\text{syn}}_w) \leq \mathcal{L}_{\text{ce}}(\mathbf{\hat{\theta}}^*(w); D^{\text{syn}}_w) + \sqrt{\frac{M \ln(1/\delta)}{2N}},
\]

(9)

Then we have

\[
\mathcal{L}_{\text{ce}}(\mathbf{\tilde{\theta}}^*(w); D^{\text{syn}}_w) \\
\leq \mathcal{L}_{\text{ce}}(\mathbf{\tilde{\theta}}^*(w); \tilde{D}^{\text{syn}}_w) + \sqrt{\frac{2M \ln(1/\delta)}{N}} \\
\leq \mathcal{L}_{\text{ce}}(\mathbf{\tilde{\theta}}^*(w); \tilde{D}^{\text{syn}}_w) + \sqrt{\frac{M \ln(1/\delta)}{2N}} + \epsilon \\
\leq \mathcal{L}_{\text{ce}}(\mathbf{\tilde{\theta}}^*(w); D^{\text{syn}}_w) + \sqrt{\frac{M \ln(1/\delta)}{2N}} + \epsilon \\
+ \sqrt{\frac{M \ln(1/\delta)}{2N}} + \epsilon \\
\leq \mathcal{L}_{\text{ce}}(\mathbf{\tilde{\theta}}^*(w); D^{\text{syn}}_w) + \sqrt{\frac{2M \ln(1/\delta)}{N}} + \epsilon,
\]
The first inequality because we require inequality (9) to hold uniformly for all $|\mathcal{W}|$ functions. The second inequality is because $\hat{w}$ is the $\epsilon$-approximated solution. The third inequality is applying inequality (9). The forth inequality is because $|\mathcal{W}| > 1$. Taking infimum over $w$ on the right hand side, we obtain the desired bound.

C Implementation Details

C.1 Full implementation details

We compare the baselines using GPT2-XL (Radford et al., 2019) as PLM. For text generation, we use Nucleus Sampling (Holtzman et al., 2020) with $p = 0.9$ as the decoding strategy and use GPT2-XL as the generator. For fair comparison, we use the best prompts designed by (Ye et al., 2022a) for data generation (listed on the Appendix). For task model training, we use 1-layer Bi-LSTM and DistilBERT-base as the light-weight classifiers. For LSTM, we use Adam optimizer (Kingma and Ba, 2015) with learning rate 1e-3. For DistilBERT-base, we finetune each dataset using Adam optimizer with learning rate 2e-5, and other default hyper-parameters as suggested by HuggingFace Transformers library (Wolf et al., 2019). Unless otherwise stated, we run LSTM for 5 epochs and run DistilBERT-base for 3 epochs for prediction.

During the optimization of sample weights, Adam optimizer with learning rate 2.5e-1 is used to update the sample-weights. In the inner loop, 1,000k synthetic data are used as the training data; in the outer loop, 50k synthetic samples are randomly sampled as the training data for fast iteration. We use 1-layer Bi-LSTM and DistilBERT-base as the tiny task model and run it for 1 epoch each time for fast iteration. We The bilevel procedure is iterated for 50 times for each task.

C.2 Prompts

For IMDb/SST-2/Rotten Tomatoes, we use the prompt designed by ZEROGen. For other tasks, following (Ye et al., 2022b), we manually design a series of prompts for each task, and report results on the best prompt for PROMPTING and ZEROGen framework.

References

Aritra Ghosh, Himanshu Kumar, and P Shanti Sastry. 2017. Robust loss functions under label noise for deep neural networks. In Proceedings of the AAAI conference on artificial intelligence, volume 31.
Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. Advances in neural information processing systems, 31.

Dan Hendrycks, Kimin Lee, and Mantas Mazeika. 2019. Using pre-training can improve model robustness and uncertainty. In International Conference on Machine Learning, pages 2712–2721. PMLR.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28–August 2, 2019, Volume 1: Long Papers, pages 3651–3657. Association for Computational Linguistics.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. 2018. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In International Conference on Machine Learning, pages 2304–2313. PMLR.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2020a. How can we know when language models know? CoRR, abs/2012.00955.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020b. How can we know what language models know? Trans. Assoc. Comput. Linguistics, 8:423–438.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.

Nikola Konstantinov and Christoph Lampert. 2019. Robust learning from untrusted sources. In International conference on machine learning, pages 3488–3498. PMLR.

Jan Kremer, Fei Sha, and Christian Igel. 2018. Robust active label correction. In International conference on artificial intelligence and statistics, pages 308–316. PMLR.

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

Alisa Liu, Swabha Swayamdipta, Noah A Smith, and Yejin Choi. 2022. Wanli: Worker and ai collaboration for natural language inference dataset creation. arXiv preprint arXiv:2201.05955.

Yang Liu and Hongyi Guo. 2020. Peer loss functions: Learning from noisy labels without knowing noise rates. In International Conference on Machine Learning, pages 6226–6236. PMLR.

Xingjun Ma, Hanxun Huang, Yisen Wang, Simone Romano, Sarah Erfani, and James Bailey. 2020. Normalized loss functions for deep learning with noisy labels. In International Conference on Machine Learning, pages 6543–6553. PMLR.

Xingjun Ma, Yisen Wang, Michael E Houle, Shuo Zhou, Sarah Erfani, Shutoo Xia, Sudanthi Wijewickrema, and James Bailey. 2018. Dimensionality-driven learning with noisy labels. In International Conference on Machine Learning, pages 3355–3364. PMLR.

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 The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA, pages 142–150. The Association for Computer Linguistics.

Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: Understanding rating dimensions with review text. In Proceedings of the 7th ACM Conference on Recommender Systems, RecSys ’13, page 165–172, New York, NY, USA. Association for Computing Machinery.

Yu Meng, Jiaxin Huang, Yu Zhang, and Jiawei Han. 2022. Generating training data with language models: Towards zero-shot language understanding. arXiv preprint arXiv:2202.04538.

Swaroop Mishra, Daniel Khoshabi, Chitta Baral, Yejin Choi, and Hannaneh Hajishirzi. 2021. Reframing instructional prompts to gptk’s language. CoRR, abs/2109.07830.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL’05), pages 115–124, Ann Arbor, Michigan. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. 2019. Language models as knowledge bases? 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 2463–2473. Association for Computational Linguistics.

Raul Puru, Ryan Spring, Mohammad Shoeybi, Mostofa Patwary, and Bryan Catanzaro. 2020. Training ques-
tion 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, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. 2014. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. 2018. Learning to reweight examples for robust deep learning. In *International conference on machine learning*, pages 4334–4343. PMLR.

Laria Reynolds and Kyle McDonell. 2021. Prompt programming for large language models: Beyond the few-shot paradigm. In *CHI ’21: CHI Conference on Human Factors in Computing Systems*, Virtual Event / Yokohama Japan, May 8-13, 2021, Extended Abstracts, pages 314:1–314:7. ACM.

Timo Schick and Hinrich Schütze. 2021. Generating datasets with pretrained language models. 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 6943–6951. Association for Computational Linguistics.

Amirreza Shaban, Ching-An Cheng, Nathan Hatch, and Byron Boots. 2019. Truncated backpropagation for bilevel optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1723–1732. PMLR.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. *Autoprompt: Eliciting knowledge from language models with automatically generated prompts*. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2020, Online, November 16-20, 2020, pages 4222–4235. Association for Computational 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. *Advances in neural information processing systems*, 32.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 1631–1642. ACL.

Sunil Thulasidasan, Tanmoy Bhattacharya, Jeff Bilmes, Gopinath Chennupati, and Jamal Mohd-Yusof. 2019. Combating label noise in deep learning using abstention. *arXiv preprint arXiv:1905.10964*.

Martin J Wainwright. 2019. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press.

Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. 2019. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 322–330.

Zirui Wang, Adams Wei Yu, Orhan Firat, and Yuan Cao. 2021. *Towards zero-label language learning*. *CoRR*, abs/2109.09193.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface’s transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Yilun Xu, Peng Cao, Yuqing Kong, and Yizhou Wang. 2019. *L_dmi: A novel information-theoretic loss function for training deep nets robust to label noise*. *Advances in neural information processing systems*, 32.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiatao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022a. *Zerogen: Efficient zero-shot learning via dataset generation*.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiatao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022b. *Zerogen: Efficient zero-shot learning via dataset generation*. *arXiv preprint arXiv:2202.07922*.

Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. 2019. How does disagreement help generalization against label corruption? In *International Conference on Machine Learning*, pages 7164–7173. PMLR.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Zhilu Zhang and Mert Sabuncu. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*, 31.