Increasing Diversity While Maintaining Accuracy: Text Data Generation with Large Language Models and Human Interventions

John Joon Young Chung
University of Michigan
jjyc@umich.edu

Ece Kamar
Microsoft Research
eckamar@microsoft.com

Saleema Amershi
Microsoft Research
samershi@microsoft.com

Abstract

Large language models (LLMs) can be used to generate text data for training and evaluating other models. However, creating high-quality datasets with LLMs can be challenging. In this work, we explore human-AI partnerships to facilitate high diversity and accuracy in LLM-based text data generation. We first examine two approaches to diversify text generation: 1) logit suppression, which minimizes the generation of languages that have already been frequently generated, and 2) temperature sampling, which flattens the token sampling probability. We found that diversification approaches can increase data diversity but often at the cost of data accuracy (i.e., text and labels being appropriate for the target domain). To address this issue, we examined two human interventions, 1) label replacement (LR), correcting misaligned labels, and 2) out-of-scope filtering (OOSF), removing instances that are out of the user's domain of interest or to which no considered label applies. With oracle studies, we found that LR increases the absolute accuracy of models trained with diversified datasets by 14.4%. Moreover, we found that some models trained with data generated with LR interventions outperformed LLM-based few-shot classification. In contrast, OOSF was not effective in increasing model accuracy, implying the need for future work in human-in-the-loop text data generation.

1 Introduction

Training custom natural language classification models has become easier with many tools (e.g., Huggingface[1]). However, data collection remains a costly part of model building. For example, existing open-source datasets may not be usable if they do not match the distribution of a model builder's target domain or do not contain desired labels. In such cases, the model builder may need to collect and label new data which could be costly (e.g., in terms of the time and resources to scrape data or pay people to generate or annotate new data).

Advances in generative large language models (LLMs), such as GPT-3 (Brown et al., 2020), present a novel approach for creating training data for classification models (Yoo et al., 2021; Sahu et al., 2022; Kumar et al., 2020). Model builders can prompt an LLM with the domain of texts and labels of interest and the LLM can quickly generate text data for the model builder's needs. This approach allows model builders to acquire a large amount of data even when they initially have no or few data instances. With the generated data, the model builder can train a separate affordable model (e.g., BERT (Devlin et al., 2019)) to perform the specific task.

While LLMs can directly support this classification task with few-shot learning, it might not be the best option for every model builder—some might not have enough resources (e.g., GPUs) or budget (e.g., credit for GPT-3) to run expensive models. Others might be concerned about privacy or security issues when they use LLMs from external APIs (e.g., OpenAI API). In such cases, generating data from LLMs and training custom models could be a more viable approach. Moreover, if we share generated datasets within the community, we can also benefit those who do not have access to LLMs. Lastly, we can also use generated datasets to test models. With these benefits of generating new text datasets with LLMs, the practical concern is how to generate high-quality datasets.

In this work, we investigate human-AI partnerships to efficiently create high-quality datasets with LLM-based text generation. High-quality datasets should have high diversity and coverage, informing the extent of data that the model may encounter. At the same time, the generated text should have high accuracy, being relevant to the model's target task while having accurate accompanying labels. To these ends, we first study two technical approaches

[1] https://huggingface.co/

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 575–593

to diversify text generation (Section 3): 1) logit suppression, which diversifies the generated texts by decreasing the probability of sampling tokens that have already appeared frequently in the previous generation, and 2) temperature sampling, which flattens the probability distribution of sampled tokens to pick less likely texts. From an experiment on eight classification tasks with GPT-3 as a text generator (Section 4), we found that diversification approaches can have mixed results. While increasing data diversity, these approaches can hurt accuracy in generation and similarity to the original datasets for the task.

We demonstrate that human interventions (Section 5) are the key to resolving these issues in text generation diversification. We examine human interventions of replacing inaccurate labels with accurate ones (label replacement) and filtering out-of-scope data (out-of-scope data filtering). With oracle studies (Section 6), we found that replacing all incorrect labels increased model accuracy by 14.4% when we used both logit suppression and high temperature. This performance increase brings in practical benefits—without label replacement, the average accuracy of models trained with GPT-3-generated data was lower than that of GPT-3 classification with few-shot learning, but with 180 instances label-replaced, the models trained with generated data started to outperform GPT-3 few-shot classification. Out-of-scope data filtering had limited utility in increasing model accuracy, possibly due to the negative impact of removing training instances. We discuss how human interventions can further facilitate the diversity and accuracy of text data generation.

Our contributions are:

- A methodolgy that combines LLM generation approaches and human supervision for diversified and accurate data generation.
- An experiment showing how text generation diversification impacts the accuracy of trained models and other qualities of the data, such as diversity and accuracy in the generation.
- Oracle studies on how human effort to replace misaligned labels and filter out-of-scope data instances can impact the performance of models trained on data generated with text diversification.

2 Related Work

2.1 Text Data Generation for Model Training

In NLP, data augmentation, where data are multiplied based on existing data, is one context where text data are generated for model training. There were many approaches, from replacing words with synonyms (Wei and Zou, 2019; Zhang et al., 2015), to randomly editing texts (Wei and Zou, 2019), predicting replaceable words (Ng et al., 2020), back-translating (Fadaee et al., 2017), generating label-flipped data (Zhou et al., 2022), or using reinforcement learning to condition generation (Liu et al., 2020). Inspired by MixUp (Zhang et al., 2018), which mixes different examples in vision data, researchers also blended texts to augment data (Guo et al., 2020; Sun et al., 2020; Zhang et al., 2022). Other approaches generate texts by learning from different datasets (Xia et al., 2020; Hou et al., 2018; Chen et al., 2020; Yoo et al., 2019).

Recently, with the generative capacity of LLMs, researchers proposed generating datasets with zero or very few samples and training a separate model to serve the specific task (Kumar et al., 2020; Yoo et al., 2021; Sahu et al., 2022; Yuan et al., 2021; Hartvigsen et al., 2022). As this approach would extract information from large models, they would be analogous to knowledge distillation (Phuong and Lampert, 2019; Hinton et al., 2015) or dataset distillation (Wang et al., 2018; Cazenavette et al., 2022). LLM-generated data has also been used to test other trained models (Ribeiro and Lundberg, 2022; Perez et al., 2022). In this work, we extend the previous work by investigating the generation of high-quality data with accurate diversification.

2.2 Text Generation with LLMs

As the size of language models increases, researchers found that LLMs can serve different generation tasks based on input prompts and examples (Brown et al., 2020). This approach can be used to generate text data with instructional prompts and a few examples. However, for the generated data to be useful, diversity and coverage should be ensured. Control of the sampling temperature (Goodfellow et al., 2016) would be relevant, as it facilitates the unlikely generation, but it was not evaluated for the facilitation of diversity and coverage. Inspired by previous work on controlling LLM generation, we examine human-AI approaches to steer data generation to have higher diversity while securing accuracy in the alignment

of specified labels.

2.3 Human-In-The-Loop

Human interventions are imperative to train high-performance machine learning models, as people curate datasets, configure model architectures, and test the trained models. Researchers investigated approaches to make human interventions more interactive in model training pipelines, by closing gaps between model training and data curation (Fogarty et al., 2008; Amershi et al., 2009, 2012; Levonian et al., 2022), humans extracting features (Branson et al., 2010; Cheng and Bernstein, 2015), interactively changing the error patterns (Kapoor et al., 2010; Talbot et al., 2009), or interactively testing models (Wu et al., 2019; Yuan et al., 2022; Ribeiro et al., 2020; Cabrera et al., 2021; Suh et al., 2019). Generative models introduce novel approaches to interactively tune and evaluate models by leveraging generated results as data instances for training and testing (Ribeiro and Lundberg, 2022). In this work, we explored harnessing diversified and accurate datasets by combining LLM-based text generation and human interventions.

3 Diversified Text Data Generation

We lay out the desired characteristics of the datasets for model building. Then, we introduce approaches to generate diversified datasets with LLMs.

3.1 Goals

Ideal classification datasets need to have the following characteristics: 1) Scoped: fall in the model builder's domain of interest while classifiable with labels of interest, 2) Label accurate: accompany accurate labels, and 3) Diverse: cover cases the model would encounter during test time. These goals are difficult to achieve simultaneously but need to be balanced. Only considering diversity, randomly generating any text would be enough, but it would hurt scope and label accuracy. Likewise, only considering the scope and label accuracy, generating an accurate but limited variety of text would be enough, but it would hurt the diversity.

3.2 Diversifying Approaches

We introduce the setting to use LLM-based data generation for model training. Then, we lay out two approaches to promote diversity in text data generation. We also note their potential risks of harming the scope and accuracy.

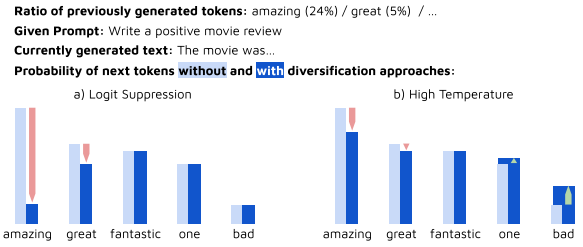

Figure 1: Examples of Diversification Approaches.

3.2.1 Settings for Data Generation

When prompting LLMs, we consider 1) a text type and 2) labels in the prompts. While there can be many different prompts, in our paper, we used the following prompt:

> Write **a movie review (`text type`)** to cover all following elements
> Elements: **positive sentiment (`label`)**
> **Movie review (`text type`)**: "This is a great movie" (A)

Model builders can also prepend examples in the same format. The generation process is iterative, and model builders can use intermediate data points as examples in later prompts. The model builders can generate data until they reach the desired number of data points. With the generated data, the model builder would finetune a separate smaller model that serves the target task. With this approach of finetuning a smaller model, there can be a question of whether finetuning a separate model would result in higher accuracy than using zero-shot or few-shot learning of the LLM. In the later study, we show the cases where finetuned smaller models perform better than the LLM.

3.2.2 Logit Suppression

Logit suppression is a diversification approach that suppresses tokens that have already been generated frequently in the intermediate dataset (Figure 1a). With this approach, the generation pipeline logs the frequency of tokens that have been generated so far. Then, to diversify the selection of tokens, logit suppression decreases the probability of high-frequency tokens. However, with this approach, some tokens that could contribute to accurate generation can be suppressed.

3.2.3 High Temperature

The temperature of sampling distribution (Goodfellow et al., 2016) controls how "flat" the token sampling probability is (the equation is explained in Appendix A). High temperature leads to "flatter" token sampling probabilities (Figure 1b), increasing the probability of sampling "less likely" tokens

and diversifying generation. Similar to logit suppression, extremely high temperatures can result in tokens irrelevant to the prompt, hurting accuracy in generation results.

4 Experiment1: Diversified Text Data Generation

We evaluated how diversification approaches impact the diversity of the generated data and the accuracy of models trained with the dataset.

4.1 Experiment Settings

4.1.1 Tasks

We used tasks from eight datasets. **SST-2** (Socher et al., 2013) is a binary sentiment classification dataset from Rotten Tomatoes movie reviews. Clickbait classification dataset (**CB**) (Chakraborty et al., 2016) is news headlines labeled either clickbait or non-clickbait. **CARER** (Saravia et al., 2018) is Twitter statements labeled with one of the six emotion categories. **PubMed** 200k RCT (Dernoncourt and Lee, 2017) has five classes regarding the roles of sentences in medical papers. The subjectivity dataset (**SUBJ**) is movie review texts labeled subjective or objective (Pang and Lee, 2004). Formality classification dataset (**FO**) (Lahiri, 2015) has labels on whether the text is formal or informal. **HWU64** (Liu et al., 2021) is a dataset with human utterances to chatbots, and we used 18 domain classes for our experiments. Corpus of Linguistic Acceptability (**COLA**) (Warstadt et al., 2019) is publication texts with annotations on whether the text is grammatically correct or not.

4.1.2 Generation Method

As a generative LLM, we used the `text-davinci-002` model of GPT-3 through OpenAI API Access with Prompt A. We list the specific text types and labels used for each dataset in Appendix B.1. The generation process was iterative, with 20 data points generated with a single prompt for each API call. As a single prompt can only generate data instances for a single label, the generation process cycled through all considered labels while balancing the number of instances for each class. As our tasks dealt with short text data, we limited the generation length to 100 tokens. We set the frequency penalty and top p to 0.02 and 1, respectively. Except for SST-2, we generated 5600 instances for a single training dataset. For SST-2, we generated 6922 data points. We chose these numbers to ensure a low generation budget while having fair quality when training models. Specifically, with a maximum length of 100 tokens for each generated instance, if the prompt includes examples for n classes, the number of required tokens for each instance would be (100+30) $\times$ (n+1) (where 30 come from the instructional prompts). With the generation pricing of \$0.02/1000 tokens for `text-davinci-002` model, 5600 and 6922 instances resulted in maximum spending of \$14.56 $\times$ (n+1) and \$17.80 $\times$ (n+1), respectively. In our pilot tests, model accuracy saturated after these numbers of instances. For the oracle training dataset, with which we compared the quality of the datasets, we sampled instances from the original training dataset for the task. The test dataset was sampled from the original test dataset. We provide details on how we sampled these instances in Appendix B.2.

Generation Conditions In addition to **logit suppression** and **temperature sampling**, we also consider **example seeding**, whether the generation pipeline begins with an initial set of example instances. We can use multiple approaches simultaneously (e.g., using logit suppression and temperature sampling together), and how these approaches interact is also the scope of our questions. For a single combination of conditions, we generated three datasets, as there could be some variance in the results with the initial seeds and the examples generated initially.

We instantiated **logit suppression** with the logit bias function in OpenAI API Access[2], which can increase or decrease the probability of sampling tokens. Every time we complete a single generation iteration, we recorded the frequency of tokens generated by GPT-3. As the OpenAI API only allows 100 tokens for logit biasing, we suppressed only the 100 most appeared tokens. Specifically, for the logit bias weights, we multiplied the token appearance ratio (in percentage) by -7.5 while capping the minimum weight at –7.5. For **temperature sampling**, we used four temperature values, 0.3, 0.7, 0.9, and 1.3. When **seeding examples**, we first randomly sampled 18 examples from oracle training data with a balanced number of labels. Only for PubMed, which has five classes, we used 15 seed examples. We used sampled data points as an initial example pool. With example seeding, from the first

[2] https://beta.openai.com/docs/api-reference/completions/create#completions/create-logit_bias

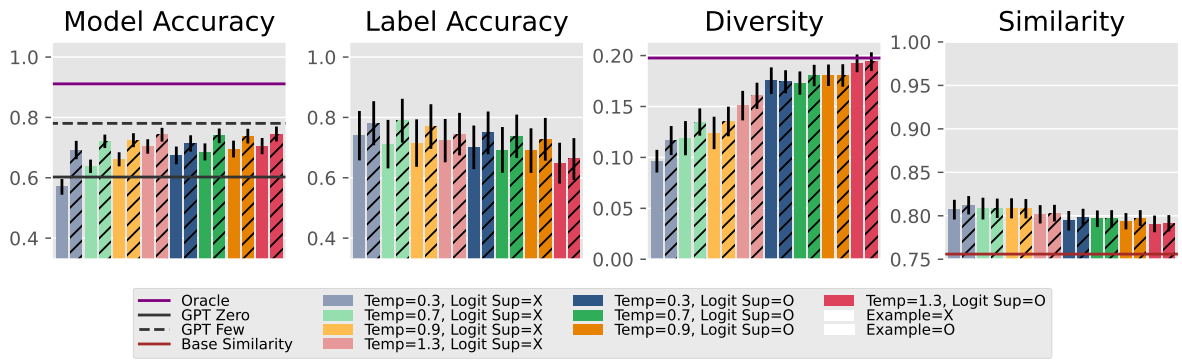

Figure 2: Impact of logit suppression and high temperatures on model accuracy, label accuracy, diversity, and similarity to the oracle dataset, averaged across eight tasks. Bars without hatches start generation without examples while those with hatches start with few-shot generation. Throughout this paper, error bars indicate 95% confidence interval.

generation iteration, examples were randomly chosen from the pool. Without the seeding examples, we completed the first cycle of generations as a zero-shot generation. After the first cycle, since we would have generated data instances for all labels, we added examples to the prompt. When adding examples, we randomly sampled the examples for all labels, one example for each label.

4.1.3 Training Method

With the generated data, we finetuned base size BERT (Devlin et al., 2019) classifiers with 109M parameters using pretrained weights from the Huggingface Transformer library (Wolf et al., 2020) with a randomly initialized fully connected classifier layer. For each dataset, we trained the five different models with the same dataset. With three datasets for each combination of approaches, it resulted in 15 models for a condition. While training, Adam optimizer was used, with a learning rate of 3e-5 and a warm-up period of 3 epochs. We adopted the early stopping with the patience of five training epochs. We used PyTorch and RTX A6000 GPUs for training.

4.2 Metrics

We compared the accuracies of models trained with generated data to 1) models trained with oracle datasets (oracle model) and 2) GPT-3's few-/zero-shot classifications (`text-davinci-002`). For GPT-3 few-shot learning, we used 18 examples (15 only for PubMed) with the same number of examples for each label. We also measured the diversity of the dataset using Remote-Clique metric (Rhys Cox et al., 2021), which is the average mean pairwise distances. Specifically, we embedded generated data with BERT (Devlin et al., 2019), then calculated the distances. We also evaluated label accuracy, which is the accuracy of the alignment between the generated texts and the specified labels. For this metric, except for SST-2, we used the oracle model as the evaluator. For SST-2, we used GPT-3 few-shot classification as the evaluator, as it has higher accuracy than the oracle model. We also measured the similarity of the generated dataset to the oracle dataset with the average mean pairwise distances between the two. For similarity, we also used BERT to embed the generated texts.

4.3 Results

Figure 2 shows the results of the first experiment for all tasks. The first column shows the model accuracy results. It also shows the accuracy of zero-shot and few-shot GPT-3 classification (gray solid and dashed line, respectively) and the model trained with the oracle training dataset (purple line). The second column shows the label accuracy, and the third column shows the diversity. The diversity plots also show the diversity of oracle datasets (purple line). The last column shows the similarity. It also shows the base similarity (brown line), which is the average distance between all the different datasets that we considered.

First, to evaluate how diversity, label accuracy, and similarity impact model accuracy, we performed a linear regression analysis. The analysis showed that label accuracy, diversity, and similarity are positively correlated with model accuracy, with significance (coef=.4797 and $p<0.001$ for label accuracy, coef=.2260 and $p<0.001$ for diversity, and coef=0.1980 and $p<0.005$ for similarity).

Regarding specific patterns, logit suppression increased diversity while hurting the label accuracy and the similarity to the oracle dataset. High temperature increased diversity and decreased label accuracy, but to a smaller degree than logit suppression. The application of each diversification approach increased the model accuracy, but when used together, the benefit did not add up. For instance, in Model Accuracy of Figure 2, each high temperature (1.3, red light bars) and logit suppression (dark blue bars) could increase the model accuracy from when using a low temperature (0.3, light blue bars). However, when using them together (dark red bars), the resulting accuracy was not much different from only using high temperatures (light red bars). It indicates that the effect of logit suppression has diminished by using high temperatures and logit suppression together. Seeding examples increases label accuracy and model accuracy. Examples also slightly increased diversity when used without logit suppression. Whether models trained with LLM-generated data would have higher accuracy than zero- or few-shot learning of LLMs depends on the task. We provide a detailed result on each task in Appendix C.

5 Human Interventions to Fix Inaccurate Text Generation

The first study shows that diversifying approaches can have mixed effects, hurting the accuracy in generation. We propose two human interventions to improve the generated data, based on issues that we found from qualitatively analyzing the generated data. The first is **label replacement (LR)**, switching the misaligned label to the correct one. The second is **out-of-scope data filtering (OOSF)**, which removes instances that are outside the domain of interest and do not match any labels (OOS instances).

While LR and OOSF might facilitate accurate generation with diversifying approaches, inspecting all data points can require a lot of effort. Hence, we propose a simple way to scale the effort of the model builder, which is training a **proxy model**. With this approach, model builders will first label a small number of data points. Then, with those labels, they will train binary classifiers as proxy models, where each learns about a single label (i.e., a label class from labels of interest or if the instance is out of scope). For unlabeled data points, proxy models can make inferences on behalf of the model builder. We introduced the specific implementation of this approach in Section 6.

6 Experiment2: Human Interventions For Diversifed Text Generation

We evaluated LR and OOSF. Except for adding LR and OOSF, we used the same tasks, datasets, training methods, and metrics as in Section 4. In this section, we focus on reporting results for two temperature values, 0.3 and 1.3. We present the results with the rest of the temperatures in Appendix E. Also, in this section, when reporting, we merged conditions with and without example seeding.

6.1 Experiment Settings

6.1.1 Label Replacement

For LR, we conducted an oracle experiment. For each task, we used the highest accuracy model as the oracle labeler. Therefore, we used oracle models as a labeler, but only for SST-2, we used GPT-3 few-shot classification as a labeler. We conducted LR on the datasets generated in experiment 1.

We had two approaches for LR: 1) do LR to all data points and 2) use proxy models with LR on partial data. For 1), we inspected all generated texts with simulated labelers and replaced labels as the labelers predicted. For 2), we sampled a set of instances from the generated dataset, applied the oracle labeler to them, and then trained proxy models with those data. Specifically, we sampled 90, 180, or 270 data instances. When training, for each class, we trained a proxy model that performs binary classification for the class. For each proxy model, the data instances labeled with the target label were used as positive instances, while the rest were used as negative instances. We applied proxy models to the uninspected data to obtain confidence scores for each label. For each class, we calculated the final score as follows:

$$S_{f,i} = S_{s,i} * w + S_{p,i} * (1 - w) \quad (1)$$

where for the class i, $S_{f,i}$ is the final score, $S_{p,i}$ is the confidence score of the proxy model, $S_{s,i}$ is if the class is specified when generating the text (1 when the class is specified, 0 otherwise), and w is the weighting constant. We considered $S_{s,i}$ as there can be a chance that the proxy model is inaccurate and the correct labels are swapped. For our experiment, we used w of 0.3. We chose the label with the highest final score as the label to be replaced.

Task	Ratio	Task	Ratio
CARER	20.56%	CB	1.39%
COLA	0.00%	FO	0.56%
HWU64	0.28%	PubMed	1.11%
SST-2	3.61%	SUBJ	3.06%

Table 1: Ratio of out-of-scope instances from 360 samples.

Task	Accuracy (std)	Task	Accuracy (std)
CARER	94.93 (2.20)	CB	100 (0.00)
SST-2	97.18 (0.89)	SUBJ	97.5 (1.04)

Table 2: OOSF proxy model performance. Note that CB only had five OOS instances, with one used for test.

For training proxy models, we trained linear support vector classifiers with a maximum iteration of 10000 while using texts embedded with BERT (Devlin et al., 2019) as input. We chose to train multiple proxy models for each class over training a single proxy model for all classes, as it tends to be more reliable in our pilots when there are many classes. As the labeling of the proxy model depends on the initial samples, for each generated dataset in experiment 1, we applied the approach five times.

6.1.2 Out-of-Scope Filtering

With OOSF, we first tried to understand how OOS instances occur. Therefore, we sampled 360 data instances for each task from the union of all the datasets generated for the task. Then, an author served as the oracle and annotated if they were OOS or not. Note that, as the definition of OOS instance, we filtered those instances that are outside the task domain or to which no label is applicable. We found that COLA, FO, HWU64, and PubMed have zero to four instances of OOS (Table 1). For the later analysis, we only considered the rest of the datasets, with at least five OOS instances. We present examples of OOS instances in Appendix D.1.

With the annotated data, we trained proxy models to annotate the instances unseen by the author, which were binary linear support vector classifiers with the maximum iteration of 10000 and BERT-embedded inputs. With the trained model, we did OOSF on the datasets generated in experiment 1. Table 2 shows the accuracy of the proxy model, when we divide the annotated data into training and test sets with an 8:2 ratio, with a split of ten times. Note that the perfect accuracy in CB is because we identified only five OOS instances from our samples, which are extremely few.

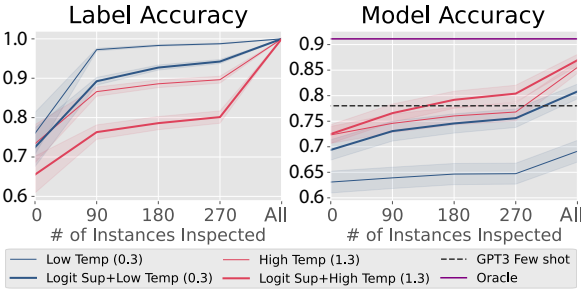

Figure 3: Impact of label replacement on label accuracy and model accuracy. Throughout this paper, error areas indicate 95% confidence interval.

After applying LR or OOSF, we trained BERT models that serve the target task. For each dataset that applied LR without proxy models or used OOSF, we ran the training five times. For each dataset that used LR with proxy models, since each dataset from experiment 1 has been label-replaced five times, we ran training only once. With this approach, we acquired 15 model accuracy results for each task and condition.

6.2 Results

6.2.1 Label Replacement

Label Accuracy and Model Accuracy in Figure 3 shows the results with LR. It shows how model accuracy and label accuracy change with the number of instances inspected (x-axis). Other metrics, diversity, and similarity would not change with LR, as it keeps the texts as they are. For model accuracy, we also visualized the performance of oracle models and the GPT-3 few-/zero-shot classification.

LR increases the model accuracy and label accuracy. Moreover, with more labels inspected, the model accuracy and label accuracy further increased. LR also added more values to logit suppression. For example, without LR, using both high temperature (1.3) and logit suppression did not have a comparative benefit over using only high temperature. However, with label replacement, the addition of logit suppression started to benefit the model accuracy when using high temperature. When doing LR with proxy models, the benefit of logit suppression increased with more instances inspected, but with full LR, the size of this gap decreased a little bit. With LR of all instances, using both high temperature and logit suppression increased the absolute model accuracy by 17.8%, compared to when using neither. It was greater than

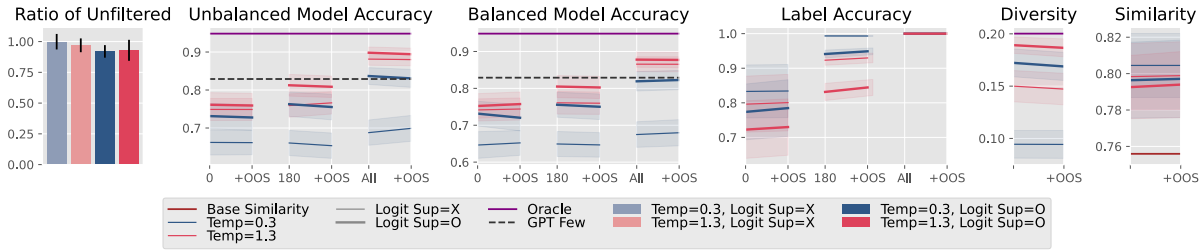

Figure 4: The ratio of instances filtered with OOSF, and its impact on model accuracy, label accuracy, diversity, and similarity, in aggregation across all tasks. As we examined the effect of OOSF with LR, for model accuracy and label accuracy, numbers left to +OOS indicate how many instances are inspected with LR.

the increase from diversification approaches when LR was not used (9.4%). Furthermore, with high temperature and logit suppression, using LR on all instances could increase the absolute model accuracy by 14.4% compared to not doing LR. When a high temperature and logit suppression are used together, the model accuracy outperformed GPT3's few-shot classification when LR was done for 180 instances. Across tasks, we found that specific patterns on how diversification approaches and LR impact the model accuracy can vary between tasks. We provide details in Appendix E.1.

6.2.2 Out-of-Scope Instances Filtering

Figure 4 shows how many instances were filtered with OOSF and how it affects model accuracy, label accuracy, diversity, and similarity. We present model accuracy from both unbalanced and balanced data: when we balanced data, we used datasets with the same number of instances across different conditions by subsampling data with the smallest size of the filtered dataset. It was because filtering can make the number of instances different between conditions. For unbalanced data, we did not balance the number of instances.

OOSF either increases or maintains label accuracy and similarity while decreasing or maintaining diversity, but there was no unified pattern of how they impact the model accuracy. There tend to be few OOS-filtered instances without diversification approaches. For example, with a temperature of 0.3 and without logit suppression, OOSF removed very few data instances. Consequently, label accuracy, diversity, and similarity remained the same with OOSF. Without diversification approaches, the accuracy of trained models tends to be more unstable with large confidence intervals. On the other hand, with diversification approaches, OOSF removed more instances, and hence there were slightly more changes in label accuracy, diversity, and similarity, with small increases in label accuracy and similarity while decreasing diversity. However, in some cases, these changes were subtle or within the 95% confidence intervals. Moreover, how the OOSF changes the model accuracy depends on the specific task and condition. We provide the OOSF results for each task in Appendix E.2.

7 Conclusion

In this work, we investigate approaches to harness LLMs and human efforts to generate text classification datasets with high accuracy and diversity. We study two text generation diversification approaches, 1) logit suppression, which restrains generating already frequently generated tokens, and 2) high temperature, which flattens the sampling probability of tokens. We found that they diversify text generation but hurt the accuracy in aligning specified labels with the generated data. We experiment with two human intervention approaches, 1) replacing misaligned labels with more adequate ones, and 2) filtering out-of-scope instances. We found that replacing labels makes diversification approaches more beneficial by increasing the accuracy of models trained with the generated dataset. On the other hand, efficient filtering of out-of-scope instances did not have a positive impact on the model accuracy.

8 Limitations

Our implementation of proxy models applies those models after the whole data is generated. Due to this, in the resulting dataset, the number of instances can often be unbalanced between labels. Such a limitation might be addressable by training proxy models from intermediate datasets with a smaller number of instances, and using those models while generating the rest of the dataset. As the data become unbalanced during the generation,

the generation pipeline can try to generate more instances with labels that are a minority in the intermediate dataset. However, when we piloted this approach, we identified potential problems. First, intermediately trained proxy models could perform worse than those trained after all data are generated, due to the lower diversity in intermediate data used to train proxy models. Second, if many data points generated with a specific label (label a) actually belong to another label (label b), there can be cases where most instances of label b come from the prompt with label a. It can skew the linguistic patterns of instances within the dataset, as only a small number of texts for label b might have been from the prompt with label b. Advanced approaches to address these issues can be future work directions.

Our implementation of efficient OOSF was not effective in increasing model accuracy. It might be due to the negative impact of removing instances, such as filtering instances on the decision boundary. As our study of OOSF was not complete, future work is necessary. Applying OOSF to the entire generated dataset and seeing the impact of their removal would be the first step. With a comprehensible understanding of OOSF, we would be able to design better OOSF strategies, such as filtering instances with various criteria.

In this work, we only examined the `text-davinci-002` model of GPT-3. Although we believe that the overall trends of results would be similar for other models, examining other models with our approaches is a necessary future work. We also examined only one prompt (Prompt A), while there may be other options. In Appendix F, we present partial results on using another prompt, showing that our approach is generalizable to other prompts. Combining human interventions with automatic annotation error detection (Klie et al., 2023) can be another future direction.

9 Ethics Statement

LLM-generated text data could have replicated biases within the used LLM. Diversification might alleviate such issues, as it steers the LLM to generate texts that it considers less probable, but bias can still exist after using the approach. More human intervention approaches can be a potential solution. For example, the model builder can provide more specific prompts and examples to counter the biased generation (Hartvigsen et al., 2022). However, these approaches still would have limitations and how these approaches would impact the data bias and the resulting model performance would need to be further researched.

Acknowledgements

We want to thank Microsoft Research for supporting the work.

References

Saleema Amershi, James Fogarty, Ashish Kapoor, and Desney Tan. 2009. Overview based example selection in end user interactive concept learning. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*, UIST '09, page 247–256, New York, NY, USA. Association for Computing Machinery.

Saleema Amershi, James Fogarty, and Daniel Weld. 2012. Regroup: Interactive machine learning for on-demand group creation in social networks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, page 21–30, New York, NY, USA. Association for Computing Machinery.

Steve Branson, Catherine Wah, Florian Schroff, Boris Babenko, Peter Welinder, Pietro Perona, and Serge Belongie. 2010. Visual recognition with humans in the loop. In *Proceedings of the 11th European Conference on Computer Vision: Part IV*, ECCV'10, page 438–451, Berlin, Heidelberg. Springer-Verlag.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, Chris 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*, volume 33, pages 1877–1901. Curran Associates, Inc.

Ángel Alexander Cabrera, Abraham J. Druck, Jason I. Hong, and Adam Perer. 2021. Discovering and validating ai errors with crowdsourced failure reports. *Proc. ACM Hum.-Comput. Interact.*, 5(CSCW2).

George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A. Efros, and Jun-Yan Zhu. 2022. Dataset distillation by matching training trajectories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. 2016. Stop clickbait:

Detecting and preventing clickbaits in online news media. In *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 9–16.

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

Justin Cheng and Michael S. Bernstein. 2015. Flock: Hybrid crowd-machine learning classifiers. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, CSCW '15, page 600–611, New York, NY, USA. Association for Computing Machinery.

Franck Dernoncourt and Ji Young Lee. 2017. PubMed 200k RCT: a dataset for sequential sentence classification in medical abstracts. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 308–313, Taipei, Taiwan. Asian Federation of Natural Language Processing.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2017. Data augmentation for low-resource neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 567–573, Vancouver, Canada. Association for Computational Linguistics.

James Fogarty, Desney Tan, Ashish Kapoor, and Simon Winder. 2008. Cueflik: Interactive concept learning in image search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, page 29–38, New York, NY, USA. Association for Computing Machinery.

Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press, Cambridge, MA, USA. `http://www.deeplearningbook.org`.

Demi Guo, Yoon Kim, and Alexander Rush. 2020. Sequence-level mixed sample data augmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5547–5552, Online. Association for Computational Linguistics.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022. ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3326, Dublin, Ireland. Association for Computational Linguistics.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network.

Yutai Hou, Yijia Liu, Wanxiang Che, and Ting Liu. 2018. Sequence-to-sequence data augmentation for dialogue language understanding. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1234–1245, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Ashish Kapoor, Bongshin Lee, Desney Tan, and Eric Horvitz. 2010. Interactive optimization for steering machine classification. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, page 1343–1352, New York, NY, USA. Association for Computing Machinery.

Jan-Christoph Klie, Bonnie Webber, and Iryna Gurevych. 2023. Annotation Error Detection: Analyzing the Past and Present for a More Coherent Future. *Computational Linguistics*, 49(1):157–198.

Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. Data augmentation using pre-trained transformer models. In *Proceedings of the 2nd Workshop on Life-long Learning for Spoken Language Systems*, pages 18–26, Suzhou, China. Association for Computational Linguistics.

Shibamouli Lahiri. 2015. Squinky! a corpus of sentence-level formality, informativeness, and implicature.

Zachary Levonian, Chia-Jung Lee, Vanessa Murdock, and F. Maxwell Harper. 2022. Trade-offs in sampling and search for early-stage interactive text classification. In *27th International Conference on Intelligent User Interfaces*, IUI '22, page 566–583, New York, NY, USA. Association for Computing Machinery.

Ruibo Liu, Guangxuan Xu, Chenyan Jia, Weicheng Ma, Lili Wang, and Soroush Vosoughi. 2020. Data boost: Text data augmentation through reinforcement learning guided conditional generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9031–9041, Online. Association for Computational Linguistics.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2021. *Benchmarking Natural Language Understanding Services for Building Conversational Agents*, pages 165–183. Springer Singapore, Singapore.

Nathan Ng, Kyunghyun Cho, and Marzyeh Ghassemi. 2020. SSMBA: Self-supervised manifold based data augmentation for improving out-of-domain robustness. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1268–1283, Online. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 271–278, Barcelona, Spain.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. 2022. Red teaming language models with language models.

Mary Phuong and Christoph Lampert. 2019. Towards understanding knowledge distillation. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5142–5151. PMLR.

Samuel Rhys Cox, Yunlong Wang, Ashraf Abdul, Christian von der Weth, and Brian Y. Lim. 2021. Directed diversity: Leveraging language embedding distances for collective creativity in crowd ideation. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA. Association for Computing Machinery.

Marco Tulio Ribeiro and Scott Lundberg. 2022. Adaptive testing and debugging of NLP models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3253–3267, Dublin, Ireland. Association for Computational Linguistics.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Gaurav Sahu, Pau Rodriguez, Issam Laradji, Parmida Atighehchian, David Vazquez, and Dzmitry Bahdanau. 2022. Data augmentation for intent classification with off-the-shelf large language models. In *Proceedings of the 4th Workshop on NLP for Conversational AI*, pages 47–57, Dublin, Ireland. Association for Computational Linguistics.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

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

Jina Suh, Soroush Ghorashi, Gonzalo Ramos, Nan-Chen Chen, Steven Drucker, Johan Verwey, and Patrice Simard. 2019. Anchorviz: Facilitating semantic data exploration and concept discovery for interactive machine learning. *ACM Trans. Interact. Intell. Syst.*, 10(1).

Lichao Sun, Congying Xia, Wenpeng Yin, Tingting Liang, Philip Yu, and Lifang He. 2020. Mixup-transformer: Dynamic data augmentation for NLP tasks. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3436–3440, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Justin Talbot, Bongshin Lee, Ashish Kapoor, and Desney S. Tan. 2009. Ensemblematrix: Interactive visualization to support machine learning with multiple classifiers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, page 1283–1292, New York, NY, USA. Association for Computing Machinery.

Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. 2018. Dataset distillation. *arXiv preprint arXiv:1811.10959*.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Tongshuang Wu, Marco Tulio Ribeiro, Jeffrey Heer, and Daniel Weld. 2019. Errudite: Scalable, reproducible,

and testable error analysis. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 747–763, Florence, Italy. Association for Computational Linguistics.

Congying Xia, Chenwei Zhang, Hoang Nguyen, Jiawei Zhang, and Philip Yu. 2020. Cg-bert: Conditional text generation with bert for generalized few-shot intent detection.

Kang Min Yoo, Dongju Park, Jaewook Kang, Sang-Woo Lee, and Woomyoung Park. 2021. GPT3Mix: Leveraging large-scale language models for text augmentation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2225–2239, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kang Min Yoo, Youhyun Shin, and Sang-goo Lee. 2019. Data augmentation for spoken language understanding via joint variational generation. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press.

Ann Yuan, Daphne Ippolito, Vitaly Nikolaev, Chris Callison-Burch, Andy Coenen, and Sebastian Gehrmann. 2021. Synthbio: A case study in faster curation of text datasets. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Jun Yuan, Jesse Vig, and Nazneen Rajani. 2022. Isea: An interactive pipeline for semantic error analysis of nlp models. In *27th International Conference on Intelligent User Interfaces*, IUI '22, page 878–888, New York, NY, USA. Association for Computing Machinery.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*.

Le Zhang, Zichao Yang, and Diyi Yang. 2022. TreeMix: Compositional constituency-based data augmentation for natural language understanding. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5243–5258, Seattle, United States. Association for Computational Linguistics.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 649–657, Cambridge, MA, USA. MIT Press.

Jing Zhou, Yanan Zheng, Jie Tang, Li Jian, and Zhilin Yang. 2022. FlipDA: Effective and robust data augmentation for few-shot learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8646–8665, Dublin, Ireland. Association for Computational Linguistics.

A Equation for Temperature Sampling

Mathematically, with the temperature T and original probability of token, p_i, the temperature sampled probability of token i, $f_T(p)_i$, would be denoted as below:

$$f_T(p)_i = \frac{p_i^{1/T}}{\Sigma_j p_j^{1/T}} \quad (2)$$

B Experiment 1 Details

B.1 Prompts Used in LLM Generation

For each task, we used prompt A with text types and labels as in Table 3. For example, for CB, a prompt can look like the below with examples:

(B)

Write a **news headline** to cover all following elements
Elements: **valid news**
News headline: "Zach Johnson Wins Sony Open"
- - - - -
Write a **news headline** to cover all following elements
Elements: **clickbait**
News headline: "10 Of The Biggest Lies We Were Told In 2015"
- - - - -
Write a **news headline** to cover all following elements
Elements: **clickbait**
News headline:"

B.2 Sampling Oracle Dataset

For the oracle dataset, if there are more than 5600 data points in the original dataset (CB, CARER, HATE, COLA, HWU64, SUBJ), we subsampled 5600 training data points. For SST2, we used all 6922 instances from the original dataset. Note that these numbers are the same as the number of generated data instances. For FO, we used the original training dataset as is (with 3622 data instances), as there are fewer than 5600 instances. For test datasets, from the same original dataset excluding instances used for the oracle dataset, we sampled 2400 data points for CB, CARER, HATE, and HWU64. For FO, COLA, SUBJ, and SST-2, we used the original test datasets as there were fewer than 2400 instances.

C Results of the Experiment 1 on Individual Dataset

Here, we introduce the result of the first experiment for individual tasks (Figure 5).

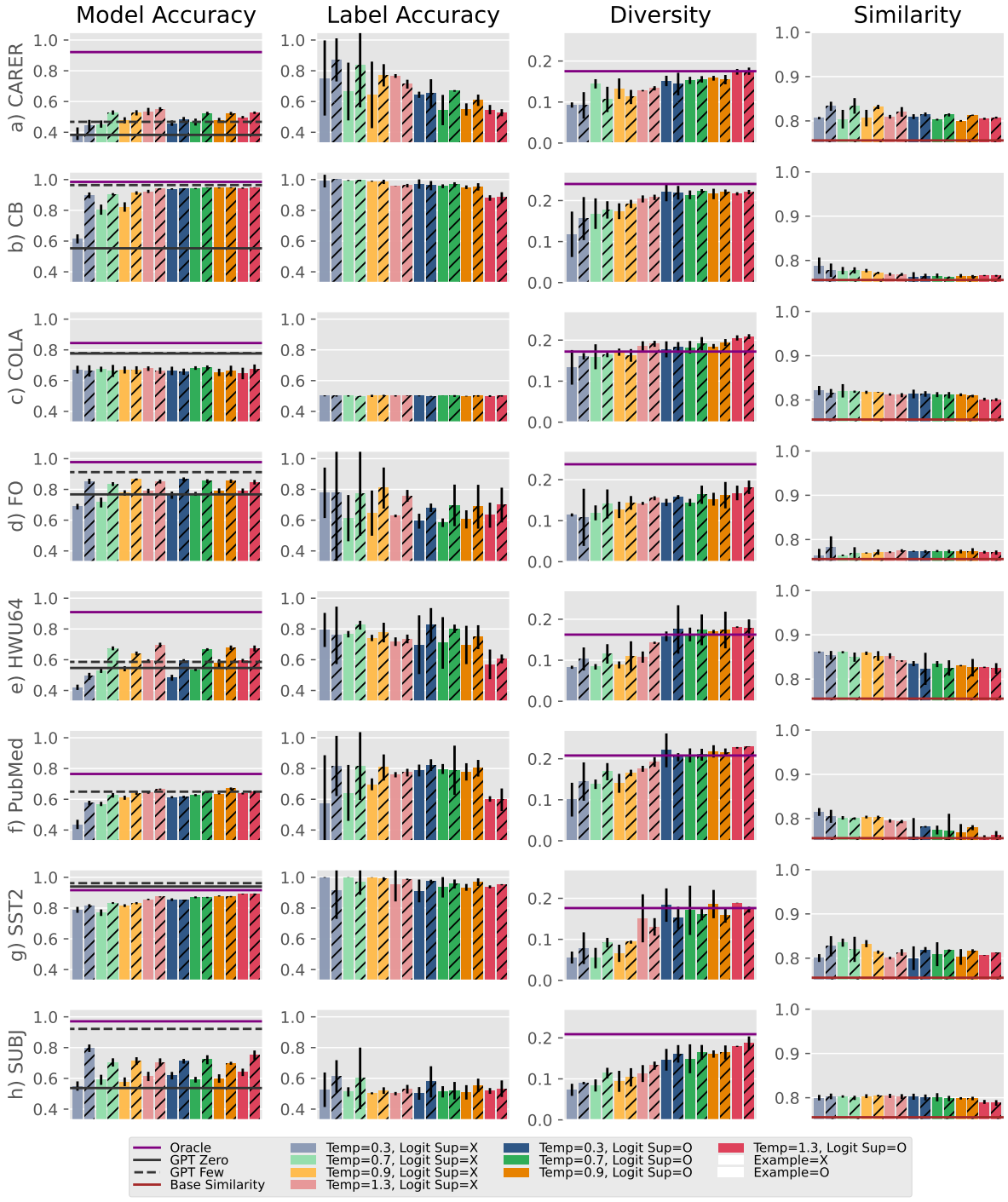

Figure 5: Impact of logit suppression and high temperatures on model accuracy, label accuracy, diversity, and similarity to the oracle dataset, for each task.

Task	Text type	Label → Label in prompts
CARER	emotional tweet	joy → expressing joy, anger → expressing anger, fear → expressing fear, sadness → expressing sadness, love → expressing love, surprise → expressing surprise
CB	news headline	non-clickbait → valid news, clickbait → clickbait
COLA	sentence	grammatically acceptable → grammatically correct sentence, grammatically unacceptable → grammatically incorrect sentence
FO	sentence	informal → informal, formal → formal
HWU64	human utterance to a chatbot	news → news, weather → weather, play → play, datetime → datetime, iot → iot, cooking → cooking, recommendation → recommendation, calendar → calendar, music → music, takeaway → takeaway, lists → list, transport → transport, qa → qa, social → social, general → general, alarm → alarm, email → email, audio → audio
PubMed	sentence from a medical paper	objective → sentence about objective, methods → sentence about methods, results → sentence about results, conclusions → sentence about conclusions, background → sentence about background
SST-2	movie review	positive → positive sentiment, negative → negative sentiment
SUBJ	sentence from a movie review	objective → objective statement, subjective → subjective statement

Table 3: Text types and labels used in prompts.

The benefit of logit suppression for each task depends on the combination of label accuracy, diversity, and similarity. Tasks that have high base label accuracy tend to improve model accuracy more with logit suppressions. For example, for CB and SST-2, those conditions with logit suppressions were clear winners in model accuracy over other combinations of approaches. For other tasks, where overall label accuracy tends to be lower, logit suppression did not have large benefits. COLA was the extreme case where the label accuracy was about 50% in binary classification, indicating that the performance of the LLM in generating label-accurate instances was not better than random chance. In this case, logit suppression resulted in almost no increase in the model accuracy. Even in this case, logit suppression could increase the diversity of the generated text. With PubMed, we could observe an exception of label accuracy increasing with logit suppression when example seeding and high temperature (1.3) are not used (compare light and dark-colored unhatched bars in PubMed's Label Accuracy from Figure 5, except for red bars). It was because GPT-3 generates many similar errors without logit suppression and seeding examples. Specifically, without logit suppression, when prompted to write about the background sentence in a medical paper, GPT-3 generated many sentences starting with "The purpose of this study was," which is more about the objective.

For temperature also, specific patterns on how it affected label accuracy, diversity, and similarity differ between tasks. In PubMed, without logit suppression and example seeding, label accuracy even increased with higher temperatures, which was against the general pattern. In this case, similar to what we found with logit suppression, the lack of diversification approaches led to the generation of narrowly populated error instances. CARER was another case with the reversed trend: without logit suppression and seeding examples, the mean diversity was higher with a temperature of 0.7 than with a temperature of 1.3. It was because, with the high temperature of 1.3, many sentences started with "I'm so," (on average 3012 occurrences) which was less the case for the lower temperatures of 0.7 and 0.9 (on average 841.5 occurrences). In CARER, when example seeding and logit suppression are not used, label accuracy was also higher with the temperature of 1.3 than with lower temperatures, although the means were within 95% confidence intervals. In this case, with lower temperatures of 0.7 and 0.9, more instances started with "No matter what," which continues with advice on what to do in emotional situations. For such cases, no label is applicable since they are not the self-expression of emotions (on average, 32 occurrences with a temperature of 1.3 and 682.7 occurrences with temperatures of 0.7 or 0.9). Note that these are examples of out-of-scope instances. Summarizing results of logit suppression and temperature sampling, these approaches increased diversity while hurting the label accuracy, but specific patterns could vary between tasks.

The utility of example seeding in label accuracy and model accuracy could also vary between tasks. For example, in the extreme case of COLA, examples did not increase label accuracy and model accuracy. How seeding examples impact the generation of data similar to the oracle dataset also

Task	Example	Reason for filtering
CARER	`No matter what life throws at you, always remember to find joy in the little things. #HappyThoughts`	Not a self-expression of emotion
CB	`Valid News`	Not a news headline
SST-2	`Jurassic World Fallen Kingdom`	Only movie title
SUBJ	`For what it's worth,`	Incomplete sentence and unable to decide subjectivity

Table 4: Examples of OOS instances.

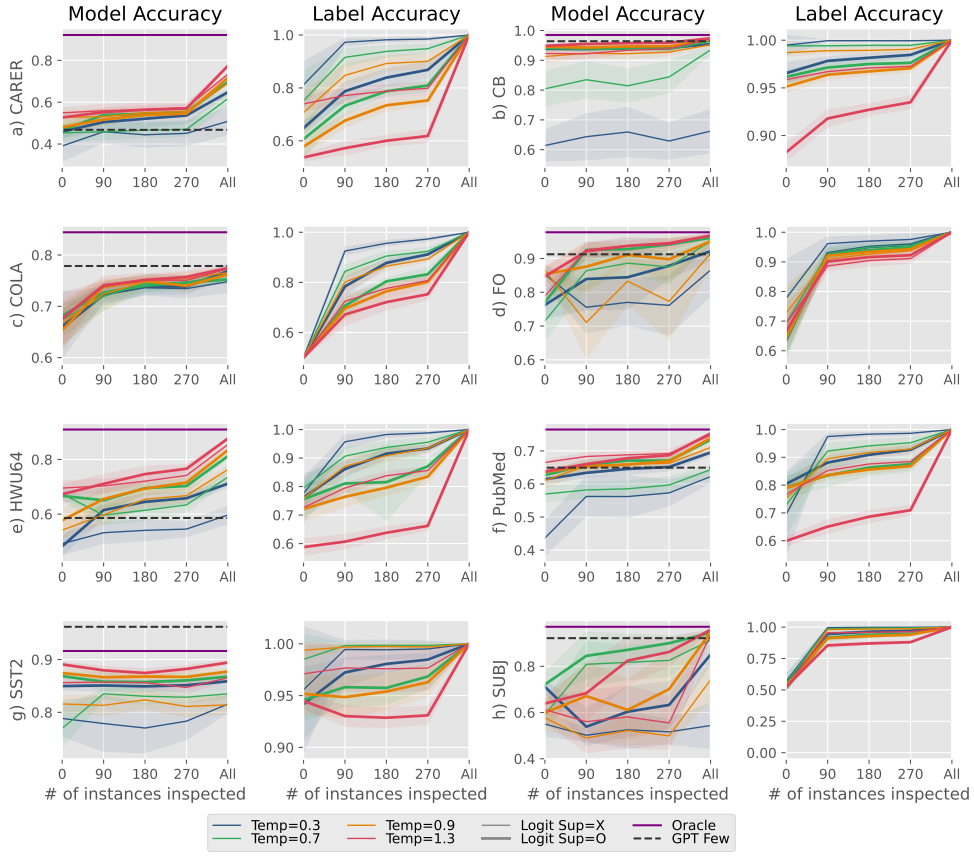

Figure 6: Impact of label replacement on model accuracy, label accuracy, for each task, on all temperature values.

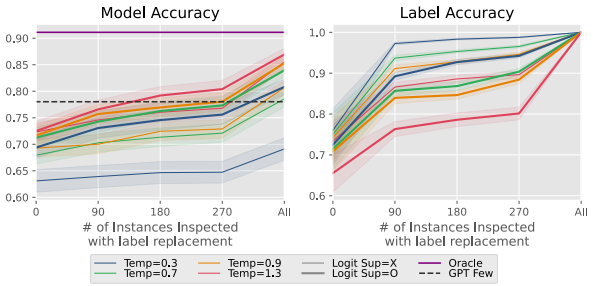

Figure 7: Impact of label replacement on model accuracy, label accuracy, for all tasks aggregated, on all temperature values.

depends on the task.

For CARER, HWU64, and PubMed in Figure 5, there were cases where the model accuracy was higher than the accuracy of GPT-3's few-shot learning. Other tasks showed lower accuracy than GPT-3's few-shot learning accuracy, indicating that GPT-3 few-shot classification can be a better alternative than training a model with generated data if the model builder has a budget to continuously access GPT-3 and is willing to hand over data through API. In Section 6, we show that human interventions can be a way to make the data generation approach applicable in more tasks by increasing the model accuracy higher than that of few-shot classifications from GPT-3.

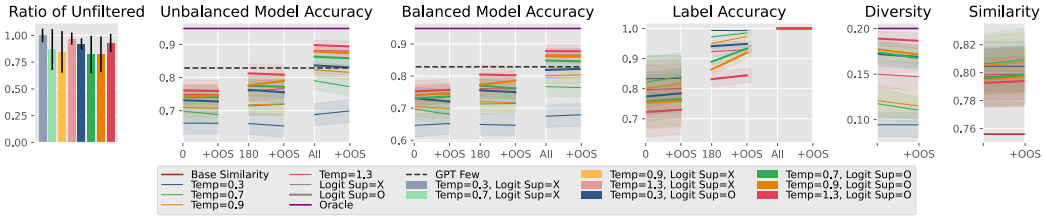

Figure 8: The ratio of instances filtered with OOSF, and its impact on model accuracy, label accuracy, diversity, and similarity, for all tasks aggregated, on all temperature values. As we examined the effect of OOSF with LR, for model accuracy and label accuracy, numbers left to +OOS indicate how many instances are inspected with LR.

D Experiment 2 Details

D.1 Examples of OOS instances.

We present examples of OOS instances in Table 4.

E Results of the Experiment 2 on Varying Tasks

We present the results of experiment 2 for individual tasks. Note that we also show results for all temperature values (0.3, 0.7, 0.9, and 1.3).

E.1 Label Replacement

Figure 6 and 7 shows the LR result for individual tasks and whole tasks aggregated, respectively, with all temperatures. First, there were cases where logit suppression provided additional benefit upon high temperature only when LR was applied (comparing thick and thin red lines in Model Accuracy of CARER, HWU64, and PubMed in Figure 6). Second, for tasks that already have high accuracy without LR (CB and SST-2), LR either resulted in very small model accuracy increases or even hurted the accuracy. For example, in SST-2, the label accuracy was already high without LR, and doing LR with proxy models could even decrease the label accuracy and model accuracy. Third, without diversification approaches, there were also cases where LR did not increase model accuracy much while label accuracy was greatly increased (thin blue lines in Model Accuracy of CARER, CB, FO, PubMed, SST2, SUBJ in Figure 6). It may show that fixing labels is more beneficial when there is enough diversity in the generated dataset. Fourth, CB, FO, and SUBJ were cases where models trained with generated data could outperform GPT-3's few-shot classification only with label replacement (some colored lines go over gray dashed lines with LR in Model Accuracy of CB, FO, and SUBJ in Figure 6). Among them, with FO, inspecting partial instances could also turn the model accuracy higher than that of GPT-3 few-shot classification. As expected, no approaches outperform oracle models as those models are used for LR. Fifth, for tasks with many classes (CARER, HWU64, and PubMed), when using LR with proxy models, the performance tends to increase not much dramatically as the number of annotated instances increases (Model Accuracy of CARER, HWU64, and PubMed in Figure 6). Higher model accuracy leaps occurred when all instances were inspected. It may indicate the difficulty of training accurate proxy models with many classes to consider.

E.2 Out-of-Scope Filtering

Figure 8 and 9 shows the OOSF results with all temperatures, for the aggregation of all tasks and individual tasks, respectively. As mentioned in the main text, it was difficult to find a general pattern of how OOSF impacts the model accuracy. Consistent patterns were that OOSF tends to increase or maintain label accuracy and similarity while decreasing or maintaining diversity.

F Results on Prompt C

On two tasks (FO, HWU64), we conducted the experiment with another instructional prompt:

> Show me a **`text type`** that has the following characteristics
> Characteristics: **`label`**
> **`text type`**: `"Generated text"` (C)

We measured model accuracy, label accuracy, diversity, and similarity of generated datasets and also investigated how label replacement impacts label accuracy and model accuracy. The experiment setting was the same as the main experiment we conducted, except the prompt used. The trend in the results (Figure 10) was similar to that of the prompt A.

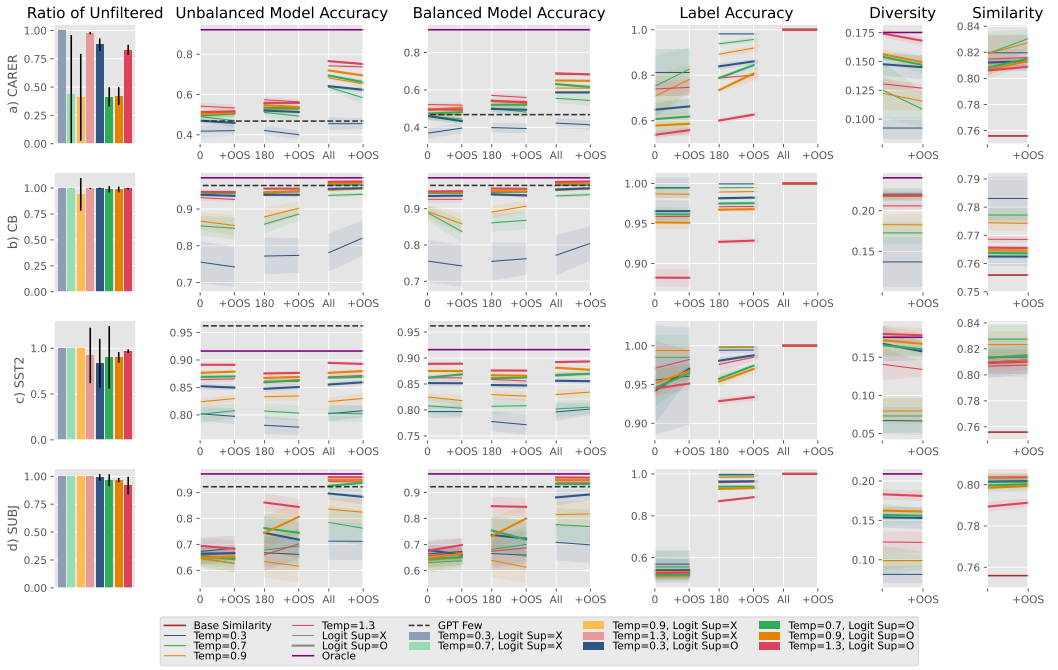

Figure 9: The ratio of instances filtered with OOSF, and its impact on model accuracy, label accuracy, diversity, and similarity, for each task, on all temperature values. As we examined the effect of OOSF with LR, for model accuracy and label accuracy, numbers left to +OOS indicate how many instances are inspected with LR.

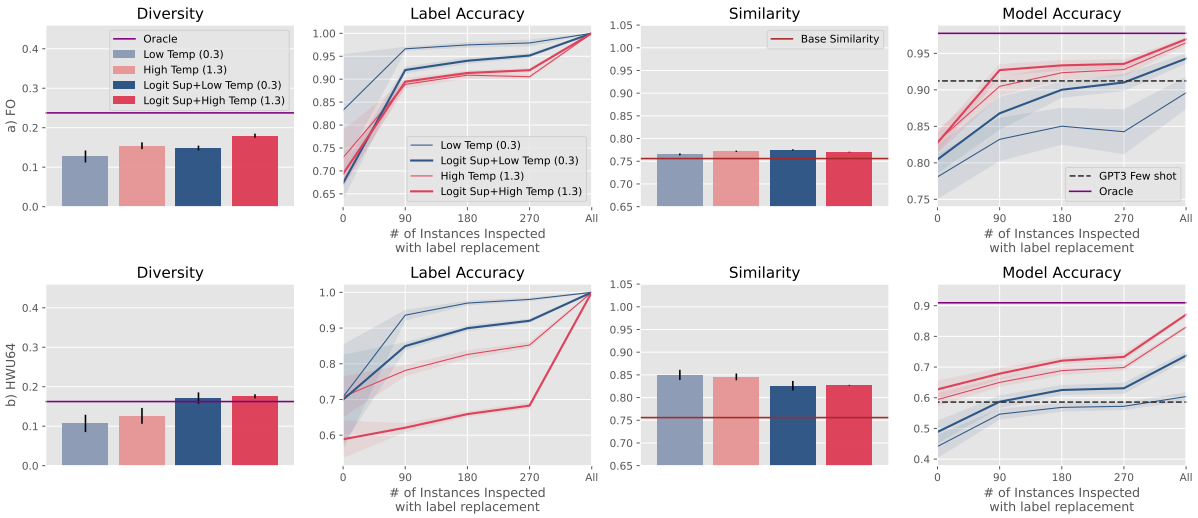

Figure 10: Result on prompt C.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 8. Limitations

☑ A2. Did you discuss any potential risks of your work?
Section 9. Ethics Statement

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Section 1. Introduction

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
Not applicable. Left blank.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Not applicable. Left blank.

C ☑ Did you run computational experiments?

Left blank.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 4 and Appendix A

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 4 and Appendix A

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Figure 1, 3 and 4 and Table 2

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 4 and Appendix A

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Section 6. It was a human oracle study where one of the authors served the role of the oracle.

☒ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
It was an oracle study by one of the authors on filtering out-of-scope instances, and we followed the definition we provided in Section 5.

☒ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
It was an oracle study by one of the authors on filtering out-of-scope instances.

☒ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
It was an oracle study by one of the authors on filtering out-of-scope instances.

☒ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
It was an oracle study by one of the authors on filtering out-of-scope instances.

☒ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
It was an oracle study by one of the authors on filtering out-of-scope instances.