Contrastive Novelty-Augmented Learning: Anticipating Outliers with Large Language Models

Albert Xu **Xiang Ren** **Robin Jia**
University of Southern California
{albertxu,xiangren,robinjia}@usc.edu

Abstract

In many task settings, text classification models are likely to encounter examples from novel classes on which they cannot predict correctly. Selective prediction, in which models abstain on low-confidence examples, provides a possible solution, but existing models are often overly confident on unseen classes. To remedy this overconfidence, we introduce Contrastive Novelty-Augmented Learning (CoNAL), a two-step method that generates OOD examples representative of novel classes, then trains to decrease confidence on them. First, we generate OOD examples by prompting a large language model twice: we prompt it to enumerate relevant novel classes, then generate examples from each novel class matching the task format. Second, we train a classifier with a novel contrastive objective that encourages lower confidence on generated OOD examples than training examples. When trained with CoNAL, classifiers improve in their ability to detect and abstain on novel class examples over prior methods by an average of 2.3% in terms of accuracy under the accuracy-coverage curve (AUAC) and 5.5% AUROC across 4 NLP datasets, with no cost to in-distribution accuracy.[1]

1 Introduction

Recent progress in NLP has led to text classification models that are accurate not only in-distribution (ID), but also on some out-of-distribution (OOD) data (Arora et al., 2021). Nonetheless, some categories of real-world distribution shift still pose serious challenges. For instance, in open-set label shift (Garg et al., 2022), the test data includes examples from novel classes not present in the training data, making it impossible for a standard classifier to predict correctly (Scheirer et al., 2013). Moreover, novel class examples can be difficult to detect with conventional OOD detection methods, as they typically bear a strong surface resemblance to training examples (Ţifrea et al., 2021). In this paper, we frame open-set label shift as a selective prediction problem (El-Yaniv and Wiener, 2010; Geifman and El-Yaniv, 2017) that we call open-set selective classification (OSSC). OSSC requires text classifiers to predict correctly on closed-set examples while abstaining on novel class examples.

To perform well on OSSC, a classifier must have lower confidence on novel class examples than closed-set examples by learning features which differentiate novel classes from closed-set classes (Perera et al., 2020). In order to supervise this representation learning, it is useful to identify what examples from novel classes might look like. Prior work has explored automatically generating OOD images by adding random perturbations to ID examples (Setlur et al., 2022). Text inputs, however, are composed of discrete tokens, and modifying even a single token can unpredictably alter the meaning of a sentence. We seek an automatic generation method that addresses these limitations, leveraging the generative ability of large language models (LLMs) like GPT-3 (Brown et al., 2020). LLMs are a desirable source for novelty, as their generation is informed by a broad corpus of examples seen during pretraining, allowing them to reliably generate from classes outside a dataset.

We present Contrastive Novelty-Augmented Learning (CoNAL), a method to improve the OSSC ability of a classifier by automatically generating OOD examples, then training to abstain on them. To generate a diverse set of OOD examples that anticipate different potential test-time shifts, we introduce Novelty Prompting, a method that augments a source dataset with novel class examples generated by a LLM. We first perform label generation, prompting our LLM to extend the closed-set labels with novel labels. We then prompt the LLM to generate new examples conditioned on each novel label to form a large set of probable novel examples.

[1]Code is available at github.com/albertkx/CoNAL.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 11778–11801

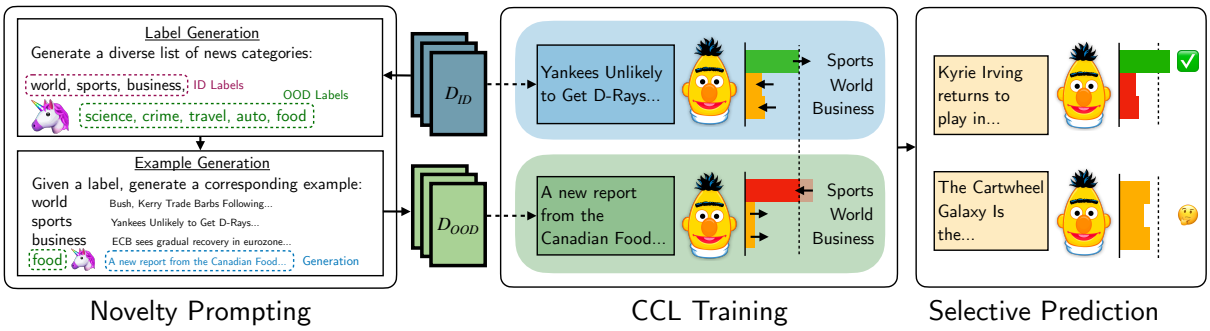

Figure 1: *Contrastive Novelty-Augmented Learning pipeline.* We Novelty Prompt a generator model to produce a novel set D_{OOD}, then train with a contrastive confidence loss (CCL) on our original train set D_{ID} and D_{OOD}, ensuring that our classifier is *less confident* on generated novel examples than closed-set examples. Finally, we abstain when the model is not very confident on any label.

Finally, we propose a contrastive confidence loss (CCL) for training, which encourages both high accuracy on the ID training set and lower relative confidence on the generated novel examples. We show that CCL outperforms stricter losses like Outlier Exposure (Hendrycks et al., 2019), which can adversely affect ID accuracy. Our full pipeline is shown in Figure 1. Our method can be viewed as a form of "partial" knowledge distillation: we leverage an LLM "teacher model" to improve novelty detection performance without altering the student model's ID classification ability.

We evaluate CoNAL against state-of-the-art OOD detection baselines across 14 splits of 4 datasets—AGNews (Zhang et al., 2015), TREC-10 (Li and Roth, 2002), TACRED (Zhang et al., 2017), and Emotion (Saravia et al., 2018)—finding that it improves both OOD detection and OSSC, by an average of 5.5% AUROC and 2.3% in terms of area under the accuracy-coverage curve (AUAC) over the best prior method. These improvements come at no cost to ID accuracy, demonstrating that it is possible to distill novelty detection alone without affecting predictive power. Finally, we analyze the settings in which CoNAL can improve OSSC performance. In the data dimension, scale is often optional: with as few as 1000 generated examples, our method outperforms vanilla training on all 4 datasets. LLM size has a larger effect on performance: on some datasets only a sufficiently large model can generate useful examples.

2 Problem Setting

2.1 Open-Set Selective Classification

In standard classification, an optimal model f should predict the ground-truth label y of an input example x from a closed set of *known* labels $\mathcal{Y}_{\text{ID}}$. However, under a more realistic *open-set* setting, some test examples are drawn from *unknown* novel classes $\mathcal{Y}_{\text{OOD}}$. Without a priori knowledge of $\mathcal{Y}_{\text{OOD}}$, a standard discriminative classifier will never correctly classify a novel example. Instead, an optimal open-set selective classifier f should predict y when $y \in \mathcal{Y}_{\text{ID}}$, and abstain otherwise.

For a probabilistic model $p_\theta(y \mid x)$ and associated confidence metric, the prediction is given by $f(x) = (\hat{y}, c)$, where $\hat{y} = \arg\max_{y \in \mathcal{Y}_{\text{ID}}} p_\theta(y \mid x)$ and c denotes the model's confidence. When used as a selective classifier with threshold γ, f predicts $\hat{y}$ when $c > \gamma$ and abstains otherwise (Geifman and El-Yaniv, 2017). This differs from OOD detection (Hendrycks and Gimpel, 2017) in that f must abstain on both novel examples and its own errors and must attain high ID accuracy.

2.2 Evaluation Protocol

We holistically measure selective classification performance with the area under the accuracy-coverage curve (AUAC). The accuracy-coverage curve plots accuracy as a function of the fraction of examples on which the model predicts (i.e., coverage) as the confidence threshold γ varies. For accuracy computation, we treat predictions on all novel class examples as incorrect. AUAC measures the combined ability of a model in ID classification accuracy, ID calibration, and OOD detection.

Though we deviate from prior work and report AUAC, to demonstrate that CoNAL is still effective at OOD detection, we also compute the Area under the ROC (AUROC). AUROC measures a model's ability to detect when a test example x is of a novel class ($y \in \mathcal{Y}_{\text{OOD}}$). Higher is better: 50% AUROC

is random, and 100% is perfect.

3 Method: CoNAL

Here we describe Contrastive Novelty-Augmented Learning, a method for automatically improving OSSC. At a high level, we generate novel examples and then train our model to be *less confident* on generated novel examples than closed-set examples. We first describe desiderata for useful novelty, then introduce a two-phased novel example generation method, Novelty Prompting, and finally introduce a contrastive confidence loss for classifier training. We illustrate the method in Figure 1.

3.1 Novelty Prompting

Desiderata of Novelty Generation Inspired by previous work which utilize known, representative OOD data to train selective prediction and OOD detection models (Kamath et al., 2020; Hendrycks et al., 2019), we focus on creating an generated "novel set" that is representative of potential label shifts at test time. The "novel set" must be (1) *plausible*, meaning that it should bear a surface resemblance to the training data, e.g., we should create news examples for a news dataset, and (2) *semantically novel*, meaning that these examples should be from new classes. In other words, an example is novel if it demonstrates a *semantic shift* (Arora et al., 2021), but shares non-semantic features with examples in the training set. For example, selecting data from an entirely separate dataset, as is done in Hendrycks et al. (2019), violates plausibility. Meanwhile simply editing surface features or recombining examples as is done in mixup (Zhang et al., 2018) might induce a distribution shift but would not result in semantic novelty.

To satisfy these desiderata, we propose a two-stage generation method called Novelty Prompting (NP). To encourage semantic novelty, we first generate novel labels given a dataset's extant labels. We then show existing examples to a language model (to encourage plausibility) and ask it to generate a new example conditioned on one of the new labels. Figure 1 shows both prompt formats.

Label Generation. Though prompting with large autoregressive language models (LLMs) like GPT-3 has typically been explored in the context of few and zero-shot learning to perform standard NLP tasks (Brown et al., 2020), we find that LLMs are also capable of "expanding" a set of topically related concepts that might realistically co-occur via sequence continuation.

We leverage this capability to generate novel labels. We prompt the largest GPT-3 model available (Davinci) with a task-specific instruction and the concatenation of the normalized known ($\mathcal{Y}_{\text{ID}}$) labels.[2] Taking the union over continuations of one or more novel labels N times, we obtain a diverse "novel label set." We combine multiple completions because in preliminary experiments, we observed that single completions tend to overgenerate labels from a narrow subcategory of classes. To remedy concerns about data leakage due to dataset examples of the true unknown class possibly appearing in LLM pretraining, we remove instances of the gold novel label(s) from this set. In practice, predicting the true novel test-time labels is both permissible and desirable, so our experimental setup likely underestimates our method's performance.

Finally, we filter out generated labels that are closely related to ID labels. For example, if `joy` appears in the ID labels, we remove synonyms like `happiness`. We use a large online thesaurus[3] to remove synonyms from the final novel label set. We analyze the impact of filtering in Appendix A.10.

Example Generation. To generate each novel example, we randomly sample a novel label from our set and prompt a LLM (we use GPT-J[4]) to generate an example of that label. We prime this model with one random sampled label-example pair from each ID class in the training dataset in the prompt, resulting in 3-6 in-context examples, varying based on the dataset. Providing these context pairs ensures that our generation is plausible: the model is encouraged to generate a specific style of text. We perform this generation procedure repeatedly to form a novel example set. We show the prompt we use for this step in Appendix A.3, and several generated label-example pairs in Figure 2.

3.2 Contrastive Confidence Loss Training

Our second contribution is an improved loss function for training models to have lower confidence on OOD examples than ID examples. Prior work have used the Outlier Exposure (OE; Hendrycks

[2]Though this requires some human intervention, it both (1) satisfies the true zero-shot nature of test-time label shift as it requires no knowledge of the unknown labels and (2) requires minimal effort, typically only involving converting an abbreviation label such as `LOC` into `Location`.

[3]`https://moby-thesaurus.org/`

[4]We evaluate GPT-3 for label generation but not example generation, as the latter would require many more API calls.

et al., 2019) objective, which encourages the model f to output a uniform probability distribution over closed-set classes when given a novel example x. OE can be successfully applied to train models on OOD data gathered from a different dataset (e.g., Wikitext), as there is very little risk of this data overlapping with ID data. In contrast, we automatically generate plausible novel examples, which runs the risk that some novel examples will be in-distribution. Since OE encourages models to have the lowest possible confidence on novel examples, it can hurt predictive accuracy when some examples x resemble closed-set examples. Instead, we seek a solution which treats outliers flexibly.

We propose a novel contrastive confidence loss (CCL) that encourages models to be less confident on OOD examples than ID examples. This is a less strict objective as models can achieve minimum loss *without* predicting a perfectly uniform distribution for the generated novel examples. For an input x, let $p_\theta(y \mid x)$ be the model's predicted distribution over $\mathcal{Y}_{\text{ID}}$. Let $c_\theta(x) = \max_{y \in \mathcal{Y}_{\text{ID}}} p_\theta(y \mid x)$, the Maximum Softmax Probability (MaxProb; Hendrycks and Gimpel, 2017), which we use as our confidence metric. Finally, let ℓ denote the cross-entropy loss with a one-hot target vector, and D_{ID} and D_{OOD} denote the training set and novel set respectively. We define CCL as follows:

$$\mathcal{L}(\theta) = \mathbb{E}_{(x_{\text{id}}, y_{\text{id}}) \sim D_{\text{ID}}}[\ell(p_\theta(y \mid x_{\text{id}}), y_{\text{id}})] + \lambda \mathbb{E}_{x_{\text{id}} \sim D_{\text{ID}}, x_{\text{ood}} \sim D_{\text{OOD}}}[\max(0, c_\theta(x_{\text{ood}}) - c_\theta(x_{\text{id}}))].$$

That is, we penalize the confidence of novel examples which have higher confidence than any closed-set example. While this still induces our model to learn lower confidence on novel examples, it simultaneously permits our model to learn that some novel examples should have lower confidence than others, rather than learn minimal confidence on *all* members of the generated novel set. In practice, we obtain an unbiased estimate of the second term by sampling a batch of n ID and n OOD examples at each step and computing the second term pairwise between each of the n^2 ID-OOD example pairs. We arbitrarily choose $\lambda = 1.0$, weighting the two terms of the objective equally.

4 Experimental Setup

4.1 Datasets

We construct artificial dataset splits from 4 popular NLP classification datasets by holding out one or more labels from training and moving all examples of that label to the test split, removing classes that are too small to yield statistical significance in our evaluations. Specifically, we use a question intent detection dataset, TREC-10 (Li and Roth, 2002) and construct 5 splits. We also use two popular topic classification datasets, AGNews (Zhang et al., 2015), a news classification dataset, and Emotion (Saravia et al., 2018), a tweet classification dataset. We construct 4 splits for each. Finally, we use TACRED (Zhang et al., 2017), a strongly class-imbalanced sentence relation-classification dataset with 41 possible relations. We construct a single split where we hold out the 35 smallest classes. Appendix A.8 contains further dataset details. Results for each dataset are averaged across all splits.

4.2 Experimental Details

For Novelty Prompting, we perform label generation using the best available GPT-3 model, GPT-3 Davinci (Brown et al., 2020) and example generation with a smaller GPT-J 6B model (Komatsuzaki, 2021). For the novel set, we perform 5 label generation iterations, then generate 100,000 examples (after filtering). We train BERT-base classifiers with CCL for 5000 steps and batch size $n = 40$. On TACRED, we permit only generations containing exactly two entities, one a subject and the other an object, filtering out roughly 90% of generations, as this is a hard constraint for relation extraction. We detail datasets in Appendices A.8 and A.9.

4.3 Baselines

We evaluate our method against baselines from prior work, CCL baselines with other novel sets, and Outlier Exposure (Hendrycks et al., 2019). Though two methods (kFolden and Constrative) can address arbitrary distribution shifts, we evaluate them here only on the open-set shift setting. For all methods, we train a BERT-base model and use hyperparameters from the original papers unless otherwise specified. Of the baselines, only CCL and Outlier Exposure use explicit novel sets.

Vanilla. We evaluate vanilla cross-entropy loss training, calculating confidence using MaxProb.

kFolden. We evaluate kFolden (Li et al., 2021), a method that trains an ensemble of k individual classifiers, each trained on $k-1$ labels. The average of the ensemble probability distributions is used for confidence computation.

Contrastive. We evaluate Contrastive OOD Detection (Zhou et al., 2021), which uses a contrastive

objective to induce training examples of different classes to be distant and of the same class to be near. This sparsifies the embedding space, ensuring that most OOD examples are far from feature representations of ID samples. We use the supervised constrastive loss and the Mahalanobis distance metric for confidence computation, finding that this setup performed the best on our evaluation.

CCL + Zero/Few-Shot Data Augmentation. To measure the impact of explicitly prompting for novel labels, we generate with an identical pretrained GPT-J model, but prompt with only an instruction and one (or zero) ID training example from each class (See Appendix A.3 for the specific prompt format). Essentially, we perform example generation identically but skip label generation entirely. We perform CCL training and MaxProb inference. While some resultant generations will be useful, we expect that many will not be semantically novel, resulting in strictly worse performance.

CCL + Wikitext. To measure whether plausibility of examples impacts their usefulness for CCL, we use an entirely different dataset, Wikitext-103, as our novel set. Though these examples represent a distribution shift, they do not accurately reflect the open-set shift the classifier will encounter.

Outlier Exposure + Novelty Prompting. We pair our novel set with Outlier Exposure (OE; Hendrycks et al., 2019) as described in Section 3.2 and compute confidence with MaxProb.

5 Results

5.1 OSSC Results

CoNAL outperforms prior work. We report comparisons of CoNAL against baselines in Table 1. Broadly, we find that while baselines like kFolden and Contrastive training struggle to consistently outperform vanilla training (e.g., on TACRED), CoNAL improves selective classification over vanilla across all datasets. We outperform the best prior method (Contrastive) by 2.3% AUAC, and on three of four datasets, our method significantly outperforms *all* prior methods. Furthermore, we outperform kFolden by 3.6% AUAC despite its ensemble totaling many times the capacity of our single classifier. CoNAL also results in zero or little accuracy drop (less than 0.2 points) for all datasets. In Appendix A.4, we show full ID accuracy results for all datasets.

Other choices of novel set for CCL training can still be beneficial. Prompting with only a task-relevant instruction (zero-shot) generates sufficiently useful novel examples to slightly outperform the vanilla baseline by 1.5% AUAC. Using Wikitext as our novel set performs roughly on par with zero-shot generation: though Wikitext examples are less noisy than generations, they also tend to be less dataset-relevant. Few-shot generation, which generates more plausible examples, is outperforms all prior methods, but performs worse than Novelty Prompting on 3 of 4 datasets.

To further test the importance of novel set selection, we compare with two oracle methods. In the Gold Data setting, we use CCL with held out data of the gold novel test class(es) as a strict upper bound for both label and example generation. In the Gold Label setting, we eliminate the label generation step, performing example generation using the gold label alone. This setting is overly optimistic as we cannot know what new labels will appear at test-time.[5] CCL in the Gold Label setting slightly outperforms CoNAL, but using gold novel data can achieve much stronger OSSC.

Training loss choice matters for generated data. Although OE training with Novelty Prompting data improves OOD detection over vanilla, it sharply decreases accuracy on TREC-10 (96.6% $\rightarrow$ 71.3%) and on average by 0.6% on the other three datasets (see Appendix A.4). In contrast, we find that CCL training maintains accuracy on all settings (see Appendix A.4), as it does not enforce a uniform probability distribution on all novel set examples. CCL with both zero- and few-shot generation outperforms all prior methods, and our full CoNAL method significantly outperforms prior methods on all but one dataset. OE exhibits this issue only with generated data: when the novel set is instead sampled from held-out gold OOD data OE outperforms CCL in AUAC and AUROC, suffering only a small accuracy drop (an average of 0.4%).

We attribute this behavior to generation noise: some generated examples are similar to ID examples, and thus greatly affect the model's ID predictions when training with OE. To verify this hypothesis, we conduct an experiment where we train classifiers with synthetic novel sets formed by noising heldout OOD data with various amounts of heldout

[5] In practice, expert knowledge of the novelty we expect to see at test-time is sometimes available, and as shown in our results, can be leveraged for better performance.

	AUAC (↑)	TREC-10	AGNews	Emotion	TACRED	Average
Baselines	Vanilla	$89.2_{\pm 2.2}$	$87.9_{\pm 0.6}$	$90.3_{\pm 1.0}$	$89.6_{\pm 0.1}$	89.3
	kFolden	$\mathbf{93.5}_{\pm 0.6}$	$85.8_{\pm 1.6}$	$90.6_{\pm 0.9}$	$84.9_{\pm 3.5}$	88.7
	Contrastive	$92.0_{\pm 0.4}$	$87.0_{\pm 0.9}$	$92.2_{\pm 0.4}$	$88.8_{\pm 0.7}$	90.0
CoNAL variants and ablations	CCL + Wikitext	$91.2_{\pm 1.4}$	$88.6_{\pm 0.6}$	$92.0_{\pm 0.4}$	$89.3_{\pm 0.5}$	90.3
	CCL + Zero-Shot	$92.5_{\pm 0.8}$	$89.1_{\pm 0.4}$	$92.6_{\pm 0.2}$	$88.9_{\pm 0.4}$	90.8
	CCL + Few-Shot	$93.5_{\pm 0.3}$	$89.7_{\pm 0.3}$	$\mathbf{93.3}_{\pm 0.1}$	$90.8_{\pm 0.1}$	91.8
	OE + Wikitext	$92.6_{\pm 0.8}$	$88.9_{\pm 0.4}$	$91.6_{\pm 0.6}$	$89.8_{\pm 0.1}$	90.7
	OE + Novelty Prompting	$83.6_{\pm 0.4}$	$\mathbf{90.6}_{\pm 0.2}$	$92.4_{\pm 0.1}$	$\mathbf{91.3}_{\pm 0.3}$	89.5
Our full method	CoNAL	$\mathbf{94.3}_{\pm 0.2}$	$\mathbf{90.5}_{\pm 0.3}$	$\mathbf{93.4}_{\pm 0.1}$	$\mathbf{91.1}_{\pm 0.2}$	**92.3**
Oracle methods	CCL + Gold Label †	$94.8_{\pm 0.3}$	$91.4_{\pm 0.3}$	$93.7_{\pm 0.1}$	$91.0_{\pm 0.2}$	92.7
	CCL + Gold Data †	$96.6_{\pm 0.1}$	$93.5_{\pm 0.1}$	$94.8_{\pm 0.2}$	$94.3_{\pm 0.4}$	94.8
	OE + Gold Data †	$96.5_{\pm 0.2}$	$94.8_{\pm 0.0}$	$95.2_{\pm 0.0}$	$96.2_{\pm 0.2}$	95.7

Table 1: *OSSC Results of CoNAL.* Methods listed below CoNAL are upper bounds. All outlier exposure (OE) methods are trained on 100K outlier generations. We average over the results of 5 seeds of all splits and report standard error of the mean in subscript. We report macro-average of all datasets in the rightmost column. Oracle methods are marked with a †. We find that CoNAL significantly outperforms all prior methods on 3 of 4 datasets, and both the Novelty Prompting and CCL loss components are important for strong performance.

Label	Generated Example
CURIOSITY	i am still interested but more interested to visit the pyramids and learn more
DESPAIR	i love my friends but sometimes i feel like im not good enough
DISAPPOINTMENT	i am a human nothing is going to keep me from flying away

Figure 2: *Example novel generations for Emotion.* In this split, the gold novel label is "sadness". Though we remove the gold novel label before example generation, many generations are still relevant to this label. More generation examples are shown in Appendix A.6.

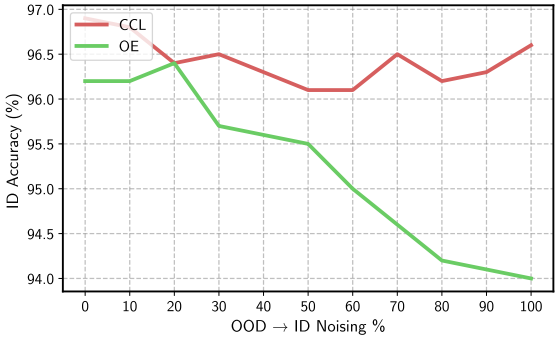

Figure 3: *Noisy novel sets hurt accuracy for OE.* We plot the ID accuracy of classifiers trained with OE and CCL on mixtures of heldout TREC-10 OOD and ID data as an novel set. ID accuracy with OE decreases as we introduce more noise, while CCL stays stable.

ID data. In Figure 3, we show that as the simulated ID noise ratio increases, OE training hurts accuracy whereas CCL models retain accuracy.

Smaller datasets suffer more. The ID accuracy drop is most salient on TREC-10 because it is by far the smallest dataset we consider, making it easy for generation noise to overwhelm signal from the train set. We conduct two experiments to show that TREC-10 is not unique, but instead exemplifies an inherent pitfall of OE. First, to test whether OE noise sensitivity applies on other datasets, we consider a smaller training set from another dataset. In the first experiment of Appendix A.12, we subsample AGNews training sets to smaller sizes. As the training set becomes smaller, the ID accuracy gap between OE and Vanilla training increases to more than 35%. Meanwhile the ID accuracy gap between CCL and Vanilla is less than 10% even at small training set sizes. Our finding here is that TREC-10 is not unique — OE can suffer from generation noise on other datasets when the training set size is not large enough.

Second, we show that this ID accuracy drop increases as the novel set size grows, i.e., the larger the novel set, the more noise the classifier sees in OE training and the worse its ID predictive ability. The second experiment in Appendix A.12 shows that when the TREC-10 novel set (100K examples) is much larger than the training set (2.8K examples), accuracy decreases drastically. We generally find a negative correlation between the novel set size and ID accuracy with OE training. In contrast, CCL maintains ID accuracy at all novel set sizes.

CCL improves ID-OOD separability. In Appendix A.13, we show that CCL is effective at OOD detection because it improves the confidence-based separability of ID and OOD examples.

	AUROC (↑)	TREC-10	AGNews	Emotion	TACRED	Average
Baselines	Vanilla	76.6$_{\pm 4.4}$	76.4$_{\pm 1.0}$	85.0$_{\pm 2.4}$	46.3$_{\pm 0.1}$	71.1
	kFolden	84.7$_{\pm 2.0}$	72.5$_{\pm 2.2}$	85.3$_{\pm 1.8}$	**53.1**$_{\pm 6.2}$	73.9
	Contrastive	79.8$_{\pm 1.3}$	76.5$_{\pm 1.8}$	89.1$_{\pm 1.7}$	45.7$_{\pm 1.2}$	72.3
CoNAL variants and ablations	CCL + Wikitext	81.0$_{\pm 2.6}$	78.1$_{\pm 0.8}$	90.3$_{\pm 0.8}$	45.2$_{\pm 1.2}$	74.1
	CCL + Zero-Shot	84.8$_{\pm 1.4}$	78.8$_{\pm 0.8}$	90.7$_{\pm 0.7}$	44.2$_{\pm 1.0}$	74.6
	CCL + Few-Shot	88.4$_{\pm 0.6}$	80.5$_{\pm 0.7}$	**92.8**$_{\pm 0.5}$	49.7$_{\pm 0.3}$	77.9
	OE + Wikitext	85.0$_{\pm 1.7}$	78.3$_{\pm 0.8}$	88.8$_{\pm 1.1}$	46.2$_{\pm 0.5}$	74.6
	OE + Novelty Prompting	74.2$_{\pm 0.5}$	**85.5**$_{\pm 0.3}$	91.0$_{\pm 0.3}$	**53.5**$_{\pm 0.7}$	76.0
Our full method	CoNAL	**90.8**$_{\pm 0.6}$	82.6$_{\pm 0.6}$	**93.4**$_{\pm 0.3}$	50.9$_{\pm 0.5}$	79.4
Oracle methods	CCL + Gold Label †	92.0$_{\pm 0.8}$	84.9$_{\pm 0.4}$	94.2$_{\pm 0.3}$	51.2$_{\pm 0.6}$	80.6
	CCL + Gold Data †	98.3$_{\pm 0.3}$	91.7$_{\pm 0.3}$	98.8$_{\pm 0.1}$	63.1$_{\pm 0.2}$	88.0
	OE + Gold Data †	99.1$_{\pm 0.2}$	98.8$_{\pm 0.3}$	99.7$_{\pm 0.0}$	89.0$_{\pm 0.5}$	96.7

Table 2: *OOD Detection Results of Contrastive Novelty-Augmented Learning.* Methods same as in Table 1. We find that CoNAL significantly improves OOD detection AUROC over all prior methods on 3 of 4 datasets. While OE training results in better AUROC on some datasets, it hurts ID accuracy.

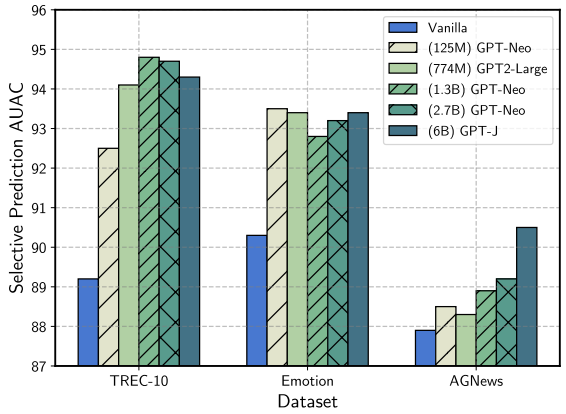

Figure 4: *Smaller generators can also help.* We perform generation with variously sized generator models, from 125M to 6B parameters. Larger generators seem to yield better results, but even our smallest generator does well.

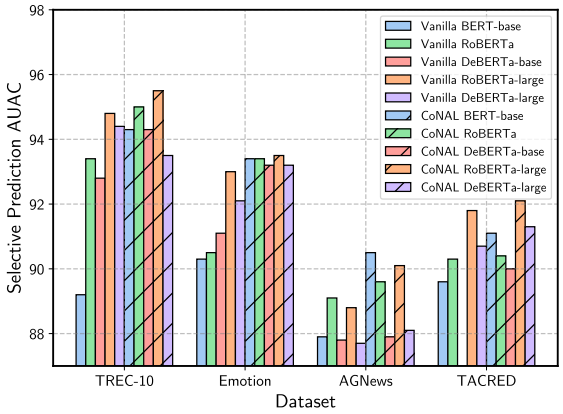

Figure 5: *Larger classifiers can also benefit.* We additionally train RoBERTa-base and RoBERTa-large with and without Novelty Prompted training. We find that both classifiers improve over vanilla with CoNAL.

5.2 OOD Detection Results

To confirm that CoNAL improves a classifier's ability to disambiguate novel class examples, we compare CoNAL against the same baselines on OOD detection in Table 2. We find similar improvements, outperforming the best prior method (kFolden) by 5.5% AUROC. We interpret this result in Appendix A.13, showing that CoNAL improves ID/OOD separability. Unlike other datasets, TACRED exhibits strong OOD overconfidence: all baselines except kFolden yield *worse*-than-random OOD detection (below 50% AUROC). We hypothesize that this could be due to models incorrectly assuming that an NER tag pair seen at training time in only a single class could not belong to a novel relation. OOD detection on TACRED remains a challenging goal for future work, as the strong performance of CCL training with gold heldout data indicates significant remaining headroom. In fact, on all three other datasets, models achieve greater than 90% AUROC when trained with gold heldout data. While OE results in better AUROC performance on AGNews, ID accuracy also decreases.

5.3 Performance Analysis

Label Generator Model We investigate whether a smaller, open-source model can suffice as the label generator. Specifically, we replace the label generator with GPT-J and use 100 label generation iterations. We find that GPT-J performs on-par with GPT-3 on 3 of 4 datasets in all metrics, except on AGNews where it performs within 1 point AUAC. We provide full details in Appendix A.5.

Example Generator Size. Since model scale often affects prompting performance (Sanh et al., 2021; Brown et al., 2020), we compare generator models ranging in size from 125M parameters to 6B parameters. For each, we generate 100K examples, and compare CoNAL results in Figure 4. All generators improve over the Vanilla baseline.

GPT2-Neo 125M is competitive with GPT2-Large despite being roughly 5x smaller, suggesting that its larger pretraining corpus (the Pile) aids generation ability. Novel generation is easier on simpler tasks: on Emotion, where labels (or synonyms) can appear directly in the example, inter-generator differences are small. We posit that even larger generators such as GPT-3 could yield better performance on abstract tasks. In Appendix A.7, we analyze the quality of generated examples.

Other Classifier Models. We investigate the generalizability of CoNAL to two other classifier architectures, RoBERTa (Liu et al., 2019) and DeBERTa-v3 (He et al., 2021), of both base and large sizes, with results in Figure 5. Averaged over datasets, CoNAL improves AUAC for all classifiers, though these improvements are most apparent with the smaller base models. Larger classifiers are better at OSSC: vanilla RoBERTa-large improves over BERT-base by 2.8% AUAC. Vanilla RoBERTa-base slightly outperforms vanilla BERT-base, but after CoNAL training, the two perform on-par, suggesting that learning from generated examples can make up for BERT's smaller pretraining corpus.

Generation Quota. Since large-scale LLM prompting is costly, we analyze the performance tradeoff of shrinking the generation quota, the number of novel examples that we can generate. In Figure 6, we show that on some datasets, using orders of magnitude smaller novel sets can still improve selective prediction. For example, 1000 generations is sufficient to improve AUAC across all datasets, and for most datasets we require far fewer. In cases where a low quota is sufficient, CoNAL is nearly as efficient as vanilla training.

Generation Analysis. To evaluate the remaining errors in OOD generation, we perform two types of manual analysis on Novelty Prompting (NP). First, we categorize the labels generated by NP after filtering, finding that 70%+ of GPT-3 generated labels are novel on all datasets except TREC-10, where only 40% are novel, and the vast majority of the others are valid closed-set labels. This highlights one source of generation noise in our pipeline. Second, we categorize the examples generated by NP and a strong baseline method, Few-shot data augmentation (FS). Specifically, for each of the 4 splits of AGNews, we annotate 100 NP and 100 FS examples. On average, 41% of NP generations come from novel classes, compared to only 26% of FS generations, explaining CoNAL's stronger performance over CCL + Few-Shot. We provide further analysis in Appendix A.7. Our method performs well despite the high fraction (50.5%) of closed-set examples generated in NP, showing that CCL is robust to noise in the example generation process.

6 Related Work

6.1 Identifying OOD Data

OOD Detection. Prior work on OOD detection uses models to detect test examples that come from a new distribution (Hendrycks and Gimpel, 2017). Many of these introduce new training objectives, e.g., with a contrastive objective (Winkens et al., 2020; Sehwag et al., 2021; Zhou et al., 2021). When the nature of the distribution shift is known, the model can directly be trained to be uncertain on known OOD examples (Dhamija et al., 2018; Hendrycks et al., 2019). We draw on the success of these known-shift methods, but eliminate the need for known OOD data by using generative models.

Other works on OOD detection have explored alternative modeling paradigms. Ensembles of neural networks can yield useful confidence estimates (Ţifrea et al., 2021; Li et al., 2021; Lakshminarayanan et al., 2017), as can simple methods like deep nearest-neighbors (Sun et al., 2022; Bergman et al., 2020). Further performance improvements can be achieved by modifying the confidence metric. Podolskiy et al. (2021) find that Mahalanobis distance better exploits the geometry of the learned embedding space, explaining strong performance achieved by replacing probability-based scoring mechanisms (Lee et al., 2018; Ren et al., 2021). We show that standard models are sufficient: MaxProb scoring with a standard classifier can perform well when given proper OOD demonstrations.

OOD Selective Prediction. Selective prediction work focuses on a different paradigm altogether, fusing abstention (detection) with prediction (El-Yaniv and Wiener, 2010; Geifman and El-Yaniv, 2017). External calibrators popularized by Kamath et al. (2020) have become popular as a selective prediction framework (Zhang et al., 2021; Ye and Durrett, 2021; Varshney et al., 2022). However, calibrators are typically smaller than classifier models (Tajwar et al., 2021); we instead update the higher-capacity classifier model to better leverage of our large set of generated outliers.

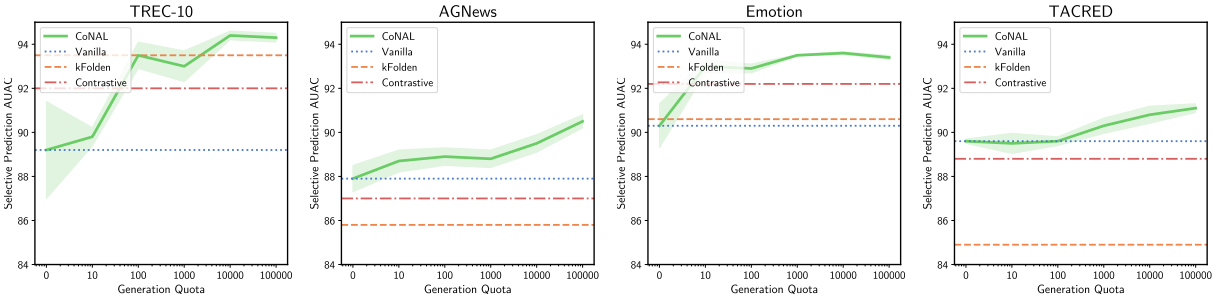

Figure 6: *Selective prediction performance is positively correlated with generation quota.* We measure selective prediction against the total number of examples generated, showing various baselines in horizontal lines. On most datasets, even a low quota can meaningfully improve AUAC.

6.2 Open-Set Classification

Open-set classification is well-explored in the image classification space, as tasks like CIFAR-100 tend towards large label spaces (Scheirer et al., 2013; Geng et al., 2021). Some methods for detecting open-set examples build on the classifier, e.g., by classifying over the model's activations (Bendale and Boult, 2016) or adding an additional reconstruction model (Oza and Patel, 2019). Our work is most closely related to methods that generate near-OOD examples and regularize confidence on them (Ge et al., 2017; Du et al., 2022; Kong et al., 2020; Vernekar et al., 2019; Möller et al., 2021; Setlur et al., 2022). However, methods like perturbation and embedding space sampling align poorly with the discrete nature of text, prompting us to investigate powerful generative language models. Esmaeilpour et al. (2022) is closely related to our work in that they also generate novel labels, but directly use these labels as input to a classifier.

Open-set classification for text has been less explored. Early works built upon the k-way, 1-vs-rest paradigm of SVMs, classifying an example as "novel" if all k scores fall below a threshold (Fei and Liu, 2016; Shu et al., 2017; Doan and Kalita, 2017). Some works explore similar methods as prior vision work, but focus on the intent detection setting, as task-oriented dialogue models should abstain on unknown intents (Zeng et al., 2021; Zheng et al., 2020; Lin and Xu, 2019). To the best of our knowledge, we are the first work to generate novel examples for open-set text classification.

6.3 Data Augmentation

Finally, our generation method, Novelty Prompting, relates to prior work in using pretrained language models for data augmentation. Kumar et al. (2021) proposes directly conditioning on class labels to generate relevant class examples, which forms a component of our prompting approach. Anaby-Tavor et al. (2020) finetunes a class-conditional generator on a given dataset to yield more relevant generations, though we consider prompting instead of finetuning as a method to prime for relevance.

7 Discussion and Future Work

In this work, we introduce CoNAL, a method for generating novel examples which simulate open-set shift and training to abstain on them. Through extensive experiments, we demonstrate that by presenting generated examples to a classifier, we can significantly improve its ability to abstain on examples from novel classes against state-of-the-art baselines. Our work provides a generalizable framework for improving OSSC and OOD detection: in fact, we show through CCL training's strong performance with gold data that there remains headroom for novel example generation. Additionally, CoNAL is modular, as it provides additional supervision signal but does not alter the classifier's architecture. It thus remains extensible with other training objectives or classification metrics. Finally, automatically diagnosing dataset issues and improving them is an important step towards making NLP safer and easier to apply. CoNAL allows practitioners to deal with noise introduced by LLM-generated data and apply these generated datasets in settings like open-set selective classification. The success of our method indicates that LLMs can be used to improve datasets with minimal human intervention. Given interest in the emergent capabilities of LLMs, we hope that future work on classification in the presence of distribution shifts can better leverage large language models to both directly identify shifts and improve the abstention ability of smaller classifiers.

Limitations

Despite the fact that we demonstrate strong OSSC performance with low generation quotas in Appendix 5.3, CoNAL still is slightly more computationally expensive than vanilla training. It also requires access to a pretrained LLM with which to generate novel examples. To achieve optimal performance, usage of the OpenAI API is required, which poses some concerns around transparency, as details around GPT-3 training and data are not publicly released. Finally, performance varies across datasets, suggesting that types of outliers that are unexpected to LLMs might still confuse a CoNAL-trained model.

Acknowledgments

We would like to thank Wang Zhu, Yuchen Lin, Muhao Chen, Wenxuan Zhou, and members of the Allegro and INK labs at USC for their valuable feedback. We also thank Johnny Wei for discussions on statistical testing.

References

Ateret Anaby-Tavor, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, N. Tepper, and Naama Zwerdling. 2020. Do not have enough data? deep learning to the rescue! In *AAAI Conference on Artificial Intelligence*.

Udit Arora, William Huang, and He He. 2021. Types of out-of-distribution texts and how to detect them. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Abhijit Bendale and Terrance E Boult. 2016. Towards open set deep networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

Liron Bergman, Niv Cohen, and Yedid Hoshen. 2020. Deep nearest neighbor anomaly detection. *arXiv preprint arXiv:2002.10445*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*.

Akshay Raj Dhamija, Manuel Günther, and Terrance E. Boult. 2018. Reducing network agnostophobia. In *Advances in Neural Information Processing Systems*.

Tri Doan and Jugal Kalita. 2017. Overcoming the challenge for text classification in the open world. In *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*.

Xuefeng Du, Zhaoning Wang, Mu Cai, and Yixuan Li. 2022. VOS: Learning what you don't know by virtual outlier synthesis. *arXiv preprint arXiv:2202.01197*.

Ran El-Yaniv and Yair Wiener. 2010. On the foundations of noise-free selective classification. In *Journal of Machine Learning Research*.

Sepideh Esmaeilpour, Bing Liu, Eric Robertson, and Lei Shu. 2022. Zero-shot out-of-distribution detection based on the pretrained model CLIP. In *Proceedings of the AAAI conference on artificial intelligence*.

Geli Fei and Bing Liu. 2016. Breaking the closed world assumption in text classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Saurabh Garg, Sivaraman Balakrishnan, and Zachary Chase Lipton. 2022. Domain adaptation under open set label shift. In *ICML 2022: Workshop on Spurious Correlations, Invariance and Stability*.

ZongYuan Ge, Sergey Demyanov, Zetao Chen, and Rahil Garnavi. 2017. Generative openmax for multi-class open set classification. *arXiv preprint arXiv:1707.07418*.

Yonatan Geifman and Ran El-Yaniv. 2017. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*.

Chuanxing Geng, Sheng-Jun Huang, and Songcan Chen. 2021. Recent advances in open set recognition: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing. In *arXiv*.

Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *5th International Conference on Learning Representations (ICLR)*.

Dan Hendrycks, Mantas Mazeika, and Thomas G. Dietterich. 2019. Deep anomaly detection with outlier exposure. In *7th International Conference on Learning Representations (ICLR)*.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2019. SpanBERT: Improving pre-training by representing and predicting spans. In *Transactions of the Association for Computational Linguistics*.

Amita Kamath, Robin Jia, and Percy Liang. 2020. Selective question answering under domain shift. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Aran Komatsuzaki. 2021. GPT-J-6B: 6B JAX-based transformer.

Lingkai Kong, Haoming Jiang, Yuchen Zhuang, Jie Lyu, Tuo Zhao, and Chao Zhang. 2020. Calibrated language model fine-tuning for in- and out-of-distribution data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2021. Data augmentation using pre-trained transformer models. In *2nd Workshop on Life-long Learning for Spoken Language Systems @ AACL 2020*.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems*.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in Neural Information Processing Systems*.

Xiaoya Li, Jiwei Li, Xiaofei Sun, Chun Fan, Tianwei Zhang, Fei Wu, Yuxian Meng, and Jun Zhang. 2021. $k$folden: k-fold ensemble for out-of-distribution detection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Ting-En Lin and Hua Xu. 2019. Deep unknown intent detection with margin loss. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

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

Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. 2019. When does label smoothing help? *Advances in Neural Information Processing Systems*.

Felix Möller, Diego Botache, Denis Huseljic, Florian Heidecker, Maarten Bieshaar, and Bernhard Sick. 2021. Out-of-distribution detection and generation using soft brownian offset sampling and autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Poojan Oza and Vishal M Patel. 2019. C2AE: Class conditioned auto-encoder for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Pramuditha Perera, Vlad I Morariu, Rajiv Jain, Varun Manjunatha, Curtis Wigington, Vicente Ordonez, and Vishal M Patel. 2020. Generative-discriminative feature representations for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Alexander Podolskiy, Dmitry Lipin, Andrey Bout, Ekaterina Artemova, and Irina Piontkovskaya. 2021. Revisiting mahalanobis distance for transformer-based out-of-domain detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Jie Ren, Stanislav Fort, Jeremiah Liu, Abhijit Guha Roy, Shreyas Padhy, and Balaji Lakshminarayanan. 2021. A simple fix to mahalanobis distance for improving near-ood detection. *arXiv preprint arXiv:2106.09022*.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. In *The Tenth International Conference on Learning Representations*.

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*.

Walter J. Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E. Boult. 2013. Toward open set recognition. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Vikash Sehwag, Mung Chiang, and Prateek Mittal. 2021. SSD: A unified framework for self-supervised outlier detection. In *The Ninth International Conference on Learning Representations*.

Amrith Setlur, Benjamin Eysenbach, Virginia Smith, and Sergey Levine. 2022. Adversarial unlearning: Reducing confidence along adversarial directions. *arXiv preprint arXiv:2206.01367*.

Lei Shu, Hu Xu, and Bing Liu. 2017. DOC: Deep open classification of text documents. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. 2022. Out-of-distribution detection with deep nearest neighbors. In *The Thirty-ninth International Conference on Machine Learning*.

Fahim Tajwar, Ananya Kumar, Sang Michael Xie, and Percy Liang. 2021. No true state-of-the-art? ood detection methods are inconsistent across datasets. In *ICML Workshop on Uncertainty & Robustness in Deep Learning*.

Neeraj Varshney, Swaroop Mishra, and Chitta Baral. 2022. Investigating selective prediction approaches across several tasks in IID, OOD, and adversarial settings. In *Findings of the Association for Computational Linguistics (ACL)*.

Sachin Vernekar, Ashish Gaurav, Vahdat Abdelzad, Taylor Denouden, Rick Salay, and Krzysztof Czarnecki. 2019. Out-of-distribution detection in classifiers via generation. In *NeurIPS 2019, Safety and Robustness in Decision Making Workshop*.

Jim Winkens, Rudy Bunel, Abhijit Guha Roy, Robert Stanforth, Vivek Natarajan, Joseph R Ledsam, Patricia MacWilliams, Pushmeet Kohli, Alan Karthikesalingam, Simon Kohl, et al. 2020. Contrastive training for improved out-of-distribution detection. *arXiv preprint arXiv:2007.05566*.

Xi Ye and Greg Durrett. 2021. Can explanations be useful for calibrating black box models? In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Zhiyuan Zeng, Keqing He, Yuanmeng Yan, Zijun Liu, Yanan Wu, Hong Xu, Huixing Jiang, and Weiran Xu. 2021. Modeling discriminative representations for out-of-domain detection with supervised contrastive learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations*.

Shujian Zhang, Chengyue Gong, and Eunsol Choi. 2021. Knowing more about questions can help: Improving calibration in question answering. In *Findings of the Association for Computational Linguistics*.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Twenty-ninth Conference on Neural Information Processing Systems*.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Yinhe Zheng, Guanyi Chen, and Minlie Huang. 2020. Out-of-domain detection for natural language understanding in dialog systems. In *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Wenxuan Zhou, Fangyu Liu, and Muhao Chen. 2021. Contrastive out-of-distribution detection for pre-trained transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Alexandru Ţifrea, Eric Stavarache, and Fanny Yang. 2021. Novelty detection using ensembles with regularized disagreement. *arXiv preprint arXiv:2012.05825*.

A Appendix

A.1 Computational Budget

As we did not closely track the total amount of computational resources used, we provide our best estimate. All experiments were completed on 12GB 1080Ti and 48GB RTX8000 GPUs. We did not perform any hyperparameter search.

We note that computation time for CoNAL is divided into two components, example generation and classifier training. We provide two examples of the computational budget for generation. When using GPT-3 as a label generator, label generation costs several cents using the `text-davinci-002` endpoint, and example generation with GPT-J 6B takes several hours on a 48GB RTX8000 GPU. We show in Appendix 5.3 that we can generate many fewer examples and still achieve strong performance, in which case generation would require orders of magnitude less time.

Classifier training for a `bert-base-cased` model takes approximately 30 minutes on a 12GB 1080 Ti GPU. For reference, vanilla training takes about half this time, as CoNAL must compute losses for a pair of batches at each training step.

A.2 Code Release

Code for replicating all experiments is released on Github at github.com/albertkx/CoNAL under an MIT license.

A.3 Prompt Format

We use the same format for label generation for all datasets, shown in Figure 7, but customize the instruction for each dataset, as shown in Figure 8.

For example generation, we prompt with an example sampled from each class and a random novel label. We use the same instruction for all datasets. An example prompt is shown in Figure 9.

Few-shot prompting is done with a task-specific instruction, but does not include labels, as shown in Figure 10. Zero-shot prompting is done with the task-specific instruction only.

A.4 Full Accuracy Results

CoNAL improves AUAC on all datasets without any cost to ID accuracy, as shown in Figure 11. We show full ID accuracy results in Table 3. CCL training maintains accuracy across all datasets, while OE training decreases accuracy on 3 of 4 datasets, with a very sharp drop on TREC-10. In Appendix A.12, we show thorough analyses on

Instruction	`Generate a diverse list of news genres:`
ID Labels	`[World, Sports, Sci/Tech,`

Figure 7: *Label Generation prompt for AGNews.*

Dataset	Instruction
Emotion	`Generate a diverse list of emotions`
AGNews	`Generate a diverse list of news genres`
TREC-10	`Generate a diverse list of entity types`
TACRED	`Generate a diverse list of relations between entities`

Figure 8: *Label Prompts for each Dataset.*

two datasets that this steep accuracy drop is not an anomaly: when paired with generated data, OE training is sensitive to the sizes of the novel set and training set, and can signficantly hurt ID accuracy when the novel set is much larger than the training set. Additionally, despite improving selective prediction performance, training with gold held-out data curiously hurts accuracy on TACRED.

A.5 CoNAL performs well without GPT-3

In our main experiments, we use GPT-3 as the label generator and GPT-J 6B as the example generator. In Section 5.3, we show that smaller models can be used as *example generators*. Here we investigate whether a smaller, open-source language model can be used as a *label generator*. In Table 12, we show that GPT-J 6B also performs well at label generation. We empirically observe that GPT-J generates shorter and noisier completions, requiring us to increase the number of model calls from 5 to 100 and filter out all labels containing punctuation marks. After applying these tweaks, we find that the difference between GPT-J and GPT-3 label generation in AUAC is small on 3 of 4 datasets, and differs by only 0.7 on AGNews, suggesting that CoNAL with GPT-J only can still work well.

A.6 Generation Examples

We show examples of the generations from Novelty Prompting for AGNews in Table 4. Recall that we do not allow the gold novel label to be generated to hedge against data leakage from LLM pretraining. However, we observe that our generator is still capable of producing relevant examples to the gold novel label due to signal from similar novel labels. Despite many generations not being directly relevant to the gold novel label, we observe that the generated novel labels are sufficiently distinct from the closed-set labels that most generated examples still provide useful "novelty" supervision signal to the classifier.

A.7 Novelty Prompting Error Analysis

Though CoNAL improves OSSC ability on all datasets, we still find headroom between Novelty Prompting generated data and gold OOD data (92.3 $\rightarrow$ 94.8) in Table 1. To understand the remaining failure modes of Novelty Prompting, we manually inspect the generated labels and examples from our method. Broadly, we seek to attribute "generation noise," or the frequency with which the purported novel sets which we generate instead contain closed-set class examples.

First, we manually annotate GPT-3 generated labels from all dataset splits, categorizing a label into "implausible" if it does conform to the dataset's format, "closed-set" (ID) if it is synonymous with a class seen in training, and "novel" (OOD) if it describes a class distinct from all closed-set classes. In Figure 13, we perform this analysis for all four datasets. Across all datasets, less than 15% of generations are implausible, suggesting that the model is usually able to generate reasonable additional labels given only 3-6 ID classes. We also observe that while on 3 of 4 datasets less than 15% of generated classes are closed-set, on TREC-10 more than half of generated labels are closed-set. One reason for this label generation noise is that the TREC-10 labels are very broad (e.g., "entity" describes questions about any subcategory of an entity, including all objects and events), so while a generated label might differ in definition, it could still overlap with or fall into a subcategory of a closed-set class.

Second, we manually annotate GPT-J generated examples to understand whether example generation is a source of generation noise. In Figure 14,

Instruction	`Given a label, generate a corresponding example:`
ID Label 1	`business`
ID Example 1	`Starwood Names New Chief Executive SEPTEMBER 21, 2004 - White Plains, NY - Former Coca-Cola Company president Steven Heyer today was named the new chief executive of Starwood Hotels, effective Oct. 1. Heyer succeeds Starwood founder Barry`
ID Label 2	`sports`
ID Example 2	`Marino, Young Considered for Hall of Fame Dan Marino and Steve Young highlighted a list Friday of 25 candidates for the Pro Football Hall of Fame.`
ID Label 3	`world`
ID Example 3	`Afghan warlords 'threaten poll' Afghan warlords are involved in intimidation which could threaten October's elections, Human Rights Watch says.`
Novel Label	`entertainment`

Figure 9: *Example Generation prompt for AGNews.*

Instruction	`Generate a news headline:`
ID Example 1	`Starwood Names New Chief Executive SEPTEMBER 21, 2004 - White Plains, NY - Former Coca-Cola Company president Steven Heyer today was named the new chief executive of Starwood Hotels, effective Oct. 1. Heyer succeeds Starwood founder Barry`
ID Example 2	`Marino, Young Considered for Hall of Fame Dan Marino and Steve Young highlighted a list Friday of 25 candidates for the Pro Football Hall of Fame.`
ID Example 3	`Afghan warlords 'threaten poll' Afghan warlords are involved in intimidation which could threaten October's elections, Human Rights Watch says.`

Figure 10: *Few-Shot Generation prompt for AGNews.*

we annotate 100 examples of each split of AGNews for both Few-shot data augmentation and Novelty Prompting. We observe that Novelty Prompting generates novel class examples more frequently across 3 of 4 splits. Both methods generate implausible (e.g., agrammatical, non-news) examples rarely, as ID demonstrations sufficiently prime the model to generate text in the style of news. Additionally, under Novelty Prompting, we find that the fraction of novel class examples (41.3%) is much lower than the fraction of novel labels generated (81.7%), suggesting that GPT-J can easily adhere to the dataset format, but struggles to extrapolate to the novel label. Future work should thus focus on better specifying the example generation step to leverage the generated labels.

A.8 Dataset Split Details

TREC-10: We remove the `Abbreviation` class as it is too small to yield statistically significant metrics in our task setting, leaving 5 remaining classes.

Emotion (Saravia et al., 2018): We remove two small classes, `love` and `surprise`, leaving 4 remaining classes.

TACRED (Zhang et al., 2017): We process the data for training following Joshi et al. (2019). This dataset is particularly challenging due to its class-imbalanced nature. We evaluate a single split where we keep the 6 largest classes as ID data, and hold out the other 35. This is the largest class, and thus results in approximately 80% of examples being OOD at test time.

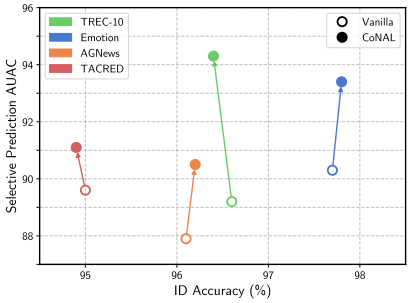

Figure 11: *CoNAL training maintains accuracy.* Training with novelty prompted examples does not significantly alter ID accuracy, but improves selective prediction across all datasets.

	(↑)	TREC-10	AGNews	Emotion	TACRED	Avg
Vanilla	AUAC	$89.2_{\pm 2.2}$	$87.9_{\pm 0.6}$	$90.3_{\pm 1.0}$	$89.6_{\pm 0.1}$	89.3
	AUROC	$76.6_{\pm 4.4}$	$76.4_{\pm 1.0}$	$85.0_{\pm 2.4}$	$46.3_{\pm 0.1}$	71.1
	ID Acc	$96.6_{\pm 0.2}$	$96.1_{\pm 0.0}$	$97.7_{\pm 0.1}$	$95.0_{\pm 0.1}$	96.4
GPT-3	AUAC	$94.3_{\pm 0.2}$	$90.5_{\pm 0.3}$	$93.4_{\pm 0.1}$	$91.1_{\pm 0.2}$	92.3
	AUROC	$90.8_{\pm 0.6}$	$82.6_{\pm 0.6}$	$93.4_{\pm 0.3}$	$50.9_{\pm 0.5}$	79.4
	ID Acc	$96.4_{\pm 0.2}$	$96.2_{\pm 0.0}$	$97.8_{\pm 0.1}$	$94.9_{\pm 0.1}$	96.3
GPT-J	AUAC	$94.2_{\pm 0.3}$	$89.8_{\pm 0.3}$	$93.5_{\pm 0.1}$	$91.0_{\pm 0.2}$	92.1
	AUROC	$90.0_{\pm 0.6}$	$80.8_{\pm 0.6}$	$93.5_{\pm 0.3}$	$50.4_{\pm 0.4}$	78.7
	ID Acc	$96.4_{\pm 0.1}$	$96.2_{\pm 0.0}$	$97.9_{\pm 0.1}$	$94.9_{\pm 0.0}$	96.4

Figure 12: *GPT-J is also a strong label generator.* We compare label generation using GPT-3 and GPT-J, using GPT-J as the example generator for both methods. GPT-J performs within a negligible margin of GPT-3 on TREC-10 and Emotion, but slightly worse on AGNews and TACRED.

Dataset	Label Type	Frequency	Example Label
TREC-10	Implausible	7.8%	August 27
	Novel	40.0%	time
	Closed-Set	52.2%	person
AGNEWS	Implausible	14.9%	ology
	Novel	81.7%	food
	Closed-Set	3.3%	technology
EMOTION	Implausible	5.1%	app
	Novel	83.9%	serenity
	Closed-Set	11.1%	frustration
TACRED	Implausible	14.3%	ualifications
	Novel	73.6%	parent company
	Closed-Set	12.1%	current location

Figure 13: *Error Analysis on Label Generation.* We manually annotate generated label sets across all splits of each dataset, recording the frequency of novel and plausible labels.

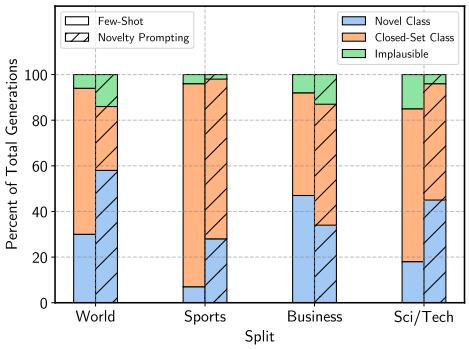

Figure 14: *Error Analysis on Example Generation.* For each split of AGNews, we manually annotate 100 generations each for two generation methods and compute the frequency of novel class examples, closed-set class examples, and implausible examples.

ID Acc (↑)	TREC-10	AGNews	Emotion	TACRED
Vanilla	$96.6_{\pm 0.2}$	$96.1_{\pm 0.0}$	$97.7_{\pm 0.1}$	$95.0_{\pm 0.1}$
kFolden	$96.5_{\pm 0.1}$	$96.0_{\pm 0.1}$	$97.2_{\pm 0.2}$	$88.3_{\pm 0.0}$
Contrastive	$95.3_{\pm 0.1}$	$96.0_{\pm 0.0}$	$98.0_{\pm 0.1}$	$94.8_{\pm 0.2}$
CCL + Wikitext	$96.6_{\pm 0.1}$	$96.1_{\pm 0.1}$	$97.6_{\pm 0.1}$	$94.9_{\pm 0.1}$
CCL + Zero-Shot	$96.5_{\pm 0.2}$	$96.3_{\pm 0.0}$	$97.6_{\pm 0.1}$	$94.9_{\pm 0.2}$
CCL + Few-Shot	$96.3_{\pm 0.2}$	$96.1_{\pm 0.0}$	$97.8_{\pm 0.1}$	$94.8_{\pm 0.2}$
OE + Wikitext	$96.6_{\pm 0.2}$	$96.1_{\pm 0.0}$	$97.6_{\pm 0.1}$	$94.8_{\pm 0.1}$
OE + Novelty Prompting	$71.3_{\pm 0.9}$	$95.6_{\pm 0.0}$	$96.4_{\pm 0.2}$	$94.8_{\pm 0.2}$
CoNAL	$96.4_{\pm 0.2}$	$96.2_{\pm 0.0}$	$97.8_{\pm 0.1}$	$94.9_{\pm 0.1}$
CCL + Gold Label †	$96.5_{\pm 0.2}$	$96.1_{\pm 0.0}$	$97.7_{\pm 0.1}$	$94.9_{\pm 0.1}$
CCL + Gold Data †	$96.0_{\pm 0.2}$	$95.8_{\pm 0.1}$	$97.6_{\pm 0.1}$	$93.8_{\pm 0.1}$
OE + Gold Data †	$96.4_{\pm 0.1}$	$95.8_{\pm 0.0}$	$97.9_{\pm 0.1}$	$93.6_{\pm 0.2}$

Table 3: *Full Accuracy Results of Contrastive Novelty-Augmented Learning*

A.9 TACRED Processing Details

We perform label normalization, removing underscores and prefixes, e.g., converting `per : employee_of` into `employee of`. This both helps the label generator model understand our label space and generate more relevant novel labels and ensures that generated novel labels are well-formatted for downstream example generation.

For examples, we normalize the Subject and Object token tags into a standard English equivalent containing the subject or object indicator and the NER tag, e.g., [`subject : person`]. To ensure that generated examples satisfy the task format, we filter out examples that do not contain exactly one subject and one object (many generations contain partial or malformed indicator/NER spans). Finally, we denormalize tags back into original model input tokens.

A.10 Label Filtering

After label generation, we perform synonym filtering to reduce occurrences of ID synonyms. We find this step to have a large impact on datasets for which labels are common English words which appear in our thesaurus, and less where label names are more abstract. For example, for Emotion and TREC-10 , where dataset names are words such as "fear" or "human," filtering removes 21% and 20% of generated labels respectively. Meanwhile on both AGNews and TACRED, label filtering removes only 2% of labels. In the case of AGNews, news genre overlaps are not easily captured by synonyms, and even after normalization, many TACRED labels such as "employee of" do not appear in our thesaurus.

A.11 Label Smoothing performs poorly.

We evaluate label smoothing (LS) (Müller et al., 2019) as an additional baseline for improving OSSC, which mirrors vanilla training but alters the one-hot target vector to a "smoother" version, incentivizing uncertainty. Label smoothing has been shown to be effective in domain shift detection (Kong et al., 2020). We use label smoothing factor $\alpha = 0.1$ and calculate confidence with MaxProb. In Table 5, we show that label smoothing performs poorly in our setting. While it does not affect classifiers' ID accuracy, it significantly decreases AUROC on all but one dataset (TREC-10), where it still remains worse than CoNAL and all of our data generation baselines.

A.12 Outlier Exposure is sensitive to generated data

In the setting where Outlier Exposure is originally evaluated, access to some known OOD data (e.g., Wikitext) is assumed (Hendrycks et al., 2019). However in our setting, where we generate a potential novel set, there is no guarantee that the generated examples are indeed OOD. For example, we show in Appendix A.7 that less than 50% of NP generations for AGNews come from novel classes. Without this guarantee, more generated data is not always better when training with OE. One risk of using more generated novel data is that the model will see a large number of ID examples in the novel

Gold Label	Generated Label	Generated Example
WORLD	HEALTH	The Top 5 Places to Take a Break From the Rat Race This summer could be the best one in years to take a break and recharge the ol' batteries.
	TRAVEL	U.S. Airlines Say They Won't Raise Rates to Pay for the New Security Fee and The government, for its part, has set a deadline for some of the other fees at the beginning of this month.
	POLITICS	The Real Reason Republicans Cautiously Endorse the 'Obscure' Paul Ryan - CNN Politics If we're going to pick fights with the other party, we might as well pick fights with the right people.
SPORTS	ENTERTAINMENT	David Beckham And Victoria Beckham Are Exiting The London Home They Have Shared With The NBA Star. LONDON (AP) - David Beckham has been kicked out of the house that he shared with his British soccer team, the Los Angeles Galaxy. The couple and the couple's kids are moving into an apartment
	FASHION	In search of fashion in South Africa What will the fashion world be wearing next year? Will the world need another white wedding dress?
	CRIME	The Police Are Going To Find You a Lot Easier in the Future: Security Advances of 2008 (Video) The police are going to find you a lot easier in the future: Security advances of 2008
BUSINESS	FOOD/DRINK	How to Make a Fried Sausage Slider (Fast) - YouTube A couple of weeks ago, I made a fried sausage slider and they're awesome! A few friends have asked me to go through this again but in a shorter time to make up a batch quickly for them.
	WEATHER	Snowstorm Blasts Northeast with Another Round of Winter Storms to Leave Snow, Ice and Blowdowns A big nor'easter is expected to bring sleet, ice and another round of dangerous wind gusts around New Year's Eve. Winter storms are expected to roll through Wednesday and Thursday in the
	RELATIONSHIPS	AP - Michael Phelps and his wife will be moving back to Washington State from Arizona while he finishes his Olympic career.
SCI/TECH	LIFESTYLE	A new batch of Apple iPhone 3Gs have gone up for sale in the UK, with all six major networks now having a network price. Apple unveiled the 3Gs on Wednesday, making a number of changes to the device, which is expected to be hugely popular in the market.
	ENTERTAINMENT	THE FILM: JERSEY GIRL "Jersey Girl" tells the story of the love-hate relationship between an Irish-American girl from New Jersey and a native New Jerseyan. Directed by Elizabeth Swados.
	TECHNOLOGY	Yahoo Japan to buy a majority stake in Nikkei Corp Yahoo Japan Corporation announced it plans to buy a 69.8 per cent stake of Nikkei for 1.43billion, the two companies said Friday.

Table 4: *Example novel generations for AGNews*.

set relative to in the training set. We conduct two experiments to analyze the impact of novel set size relative to training set size.

First, we vary the novel set size relative to the training set size. In Figure 15, we train with novel sets on TREC-10 from size 0 to 100K using both OE and CCL. We observe that training with OE hurts accuracy and AUROC when the novel set is larger than 100 examples, whereas CCL continues to improve as the novel set size grows, and maintains accuracy for all novel set sizes. As the novel set becomes larger than the size of the training set (to the right of the dashed line), both OOD detection AUROC and ID accuracy quickly decrease. This result suggests as the ID noise the classifier sees in OE training outsizes the training set, its ID predictive ability worsens.

Of the datasets in our experiments, TREC-10 is by far the smallest, with only about 2800 training examples per split. To determine whether OE is also sensitive to the size of the ID set, we subsample the AGNews dataset into smaller training sets and perform OE and CCL training with 100K-sized novel sets. We compare the results against Vanilla training with the same ID sets in Figure 16. Although reducing the training set size decreases the ID accuracy even for vanilla training, CCL training achieves similar accuracy for all subsampling sizes. We do observe that a sub-10% accuracy margin appears between vanilla and CCL at extremely small

	(↑)	TREC-10	AGNews	Emotion	TACRED	Avg
Vanilla	AUAC	$89.2_{\pm 2.2}$	$87.9_{\pm 0.6}$	$90.3_{\pm 1.0}$	$89.6_{\pm 0.1}$	89.3
	AUROC	$76.6_{\pm 4.4}$	$76.4_{\pm 1.0}$	$85.0_{\pm 2.4}$	$46.3_{\pm 0.1}$	71.1
	ID Acc	$96.6_{\pm 0.2}$	$96.1_{\pm 0.0}$	$97.7_{\pm 0.1}$	$95.0_{\pm 0.1}$	96.4
LS	AUAC	$90.6_{\pm 1.6}$	$83.5_{\pm 1.4}$	$82.0_{\pm 1.7}$	$87.1_{\pm 1.0}$	85.8
	AUROC	$80.5_{\pm 3.7}$	$72.9_{\pm 1.7}$	$75.1_{\pm 2.3}$	$41.2_{\pm 2.4}$	67.4
	ID Acc	$96.7_{\pm 0.2}$	$96.2_{\pm 0.0}$	$97.7_{\pm 0.1}$	$95.0_{\pm 0.1}$	96.4

Table 5: *Label Smoothing (LS) hurts AUROC and AUAC on all but one dataset.*

training set sizes, though this margin disappears at 1000 or more training examples. OE, meanwhile, decreases ID accuracy by as much as 35% when the dataset is subsampled to 30 examples, and 25%+ at 300 examples. OE-trained classifiers are also worse OOD detectors given limited training data: they underperform vanilla classifiers for all training sets smaller than 3000 examples. Finally, we find that OE does yield better OOD detectors than CCL for sufficiently large AGNews training sets. This expands on our findings in Table 2, suggesting that when there is access to a large amount of training data, in this case 10000 examples are more, OE can learn from noisy novel sets (though ID accuracy still decreases). Our results indicate that TREC-10 is not alone: As training set size becomes smaller, the ID classes becomes less well-specified, and ID examples present in the novel set induce the model to make incorrect predictions (and poor confidence estimates) on true ID test examples.

A.13 CoNAL and Separability

To understand why CoNAL improves AUROC, we compare the confidence profiles of a vanilla fine-tuned classifier against those of a CoNAL trained classifier. Specifically, in Figure 17, we select 50 random ID examples and 50 random OOD examples from each dataset split and compute MaxProb confidences. We find that CoNAL decreases confidence on OOD examples, though not to the same extent on all examples. In datasets like TREC-10 and Emotion where CoNAL achieves stronger AUROC gains, the decrease in OOD confidence is more pronounced. Though ID test examples also decrease in confidence on all dataset splits, this decrease is less pronounced and is likely due to the confidence contrastive objective term incentivizing the model's confidence distributions to be generally less peaked.

The shifts reflected in the confidence distributions directly impact the separability of OOD and ID examples. On the Vanilla model confidence axis, it is difficult to identify a threshold above which most examples are ID and below which most examples are OOD. Given CoNAL confidences, OOD and ID examples are more separable. This visual separability is reflected in the OOD Detection AUROC metric.

To demonstrate the strictness of the OE objective, we plot the confidences of the same examples without (Vanilla) and with OE training in Figure 18. First, we observe that the vast majority of OOD examples have similar confidence after OE training, as they are all pushed towards minimum confidence (maximum entropy). Second, we observe that OE affects the confidence of ID test examples, decreasing the confidence of some examples lower than that of OOD test examples.

A.14 Measuring Data Leakage in Generation

In our experiments, we explicitly forbid the gold novel class from being generated, such that the LLM is disincentivized from generating gold novel examples if the dataset has been seen in pretraining. However, it remains possible that if the LLM had seen the task data in pretraining, it could replicate parts of or an entire example from the dataset in generations. Unfortunately, as we do not have access to GPT-3 pretraining data, we cannot determine whether or not this is indeed a risk. Instead, we probe whether this is a possiblity via an n-gram overlap metric comparing the similarity between our generated examples and the test set.

Specifically, we measure the average fraction of n-grams in a generation that also appear in the test set, which we interpret as the maximal frequency that the LLM *could have* copied test data via pretraining leakage. For comparison, we compute the same metric between the test set and heldout novel class data. In this case, examples are sampled from exactly the same distribution and thus expected to exhibit some n-gram overlap due to shared background features. We use this value as a baseline: generation n-gram overlap should be

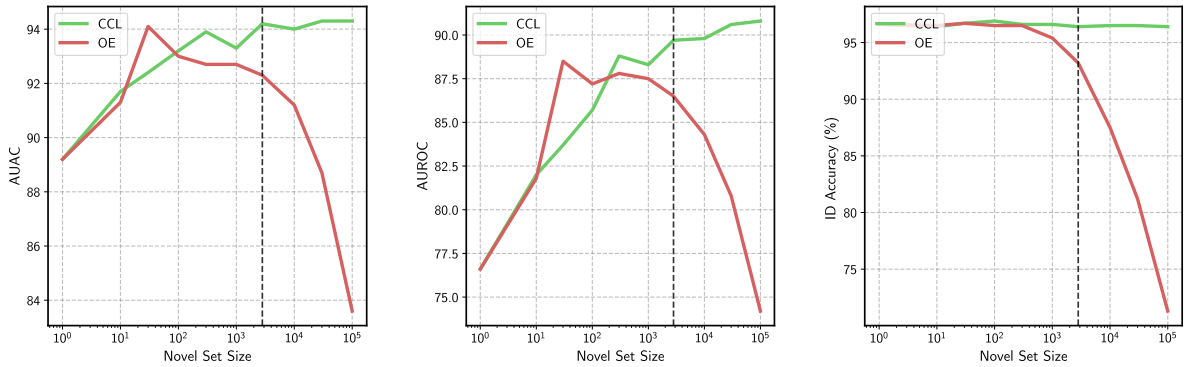

Figure 15: *Outlier Exposure is sensitive to the size of the novel set on TREC-10.* We vary the novel set size from 0 to 100K, finding that both accuracy and AUROC decrease with as few as 100 novel generations. We indicate with a dashed line the point where the novel set and training set size are approximately equal.

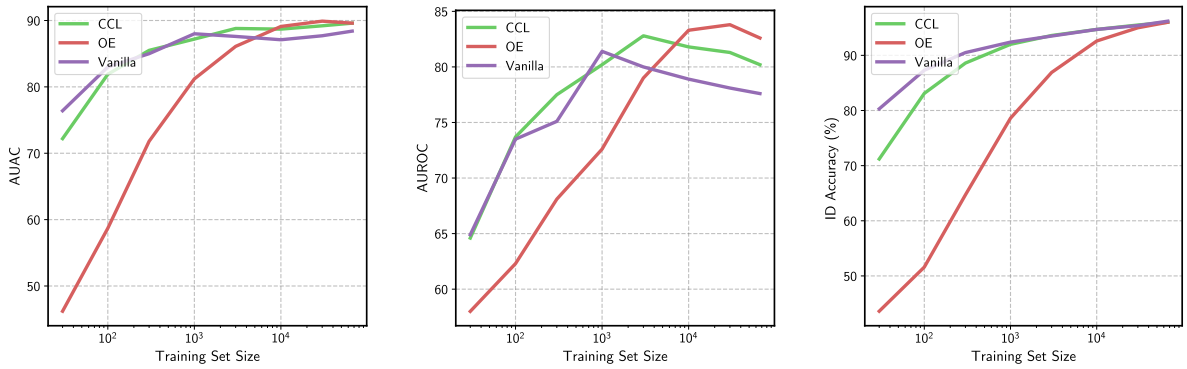

Figure 16: *Outlier Exposure disproportionately hurts smaller datasets.* We subsample the training set for AGNews, use 100K novel generated examples, and vary the training loss. We find that CCL achieves similar ID performance as Vanilla at all training set sizes, but OE hurts accuracy when the training set is smaller than 1000 examples.

similar to or lower than heldout n-gram overlap. We find in Table 6 that the n-gram overlap of our novelty prompted generations is lower across *all* datasets and values of n than of the heldout set, indicating that the performance of CoNAL should not be attributed to example data leakage.

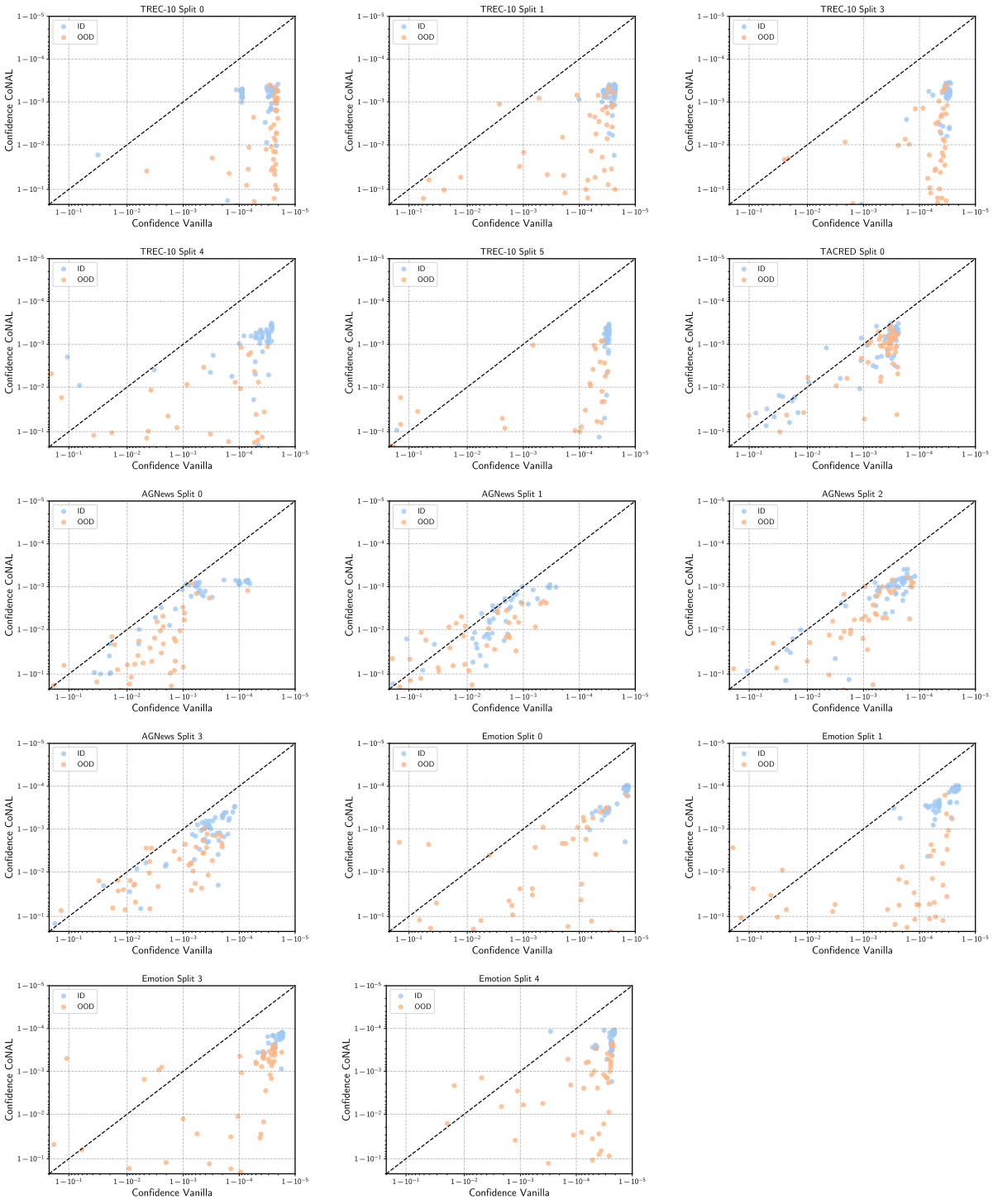

Figure 17: *CoNAL improves the separability of ID and OOD examples.* We plot the confidences of 50 random ID and 50 random OOD examples on a vanilla finetuned BERT classifier versus a CoNAL trained BERT classifier. CoNAL successfully decreases the confidence of OOD test examples while minimizing the impact of the confidence of ID test examples.

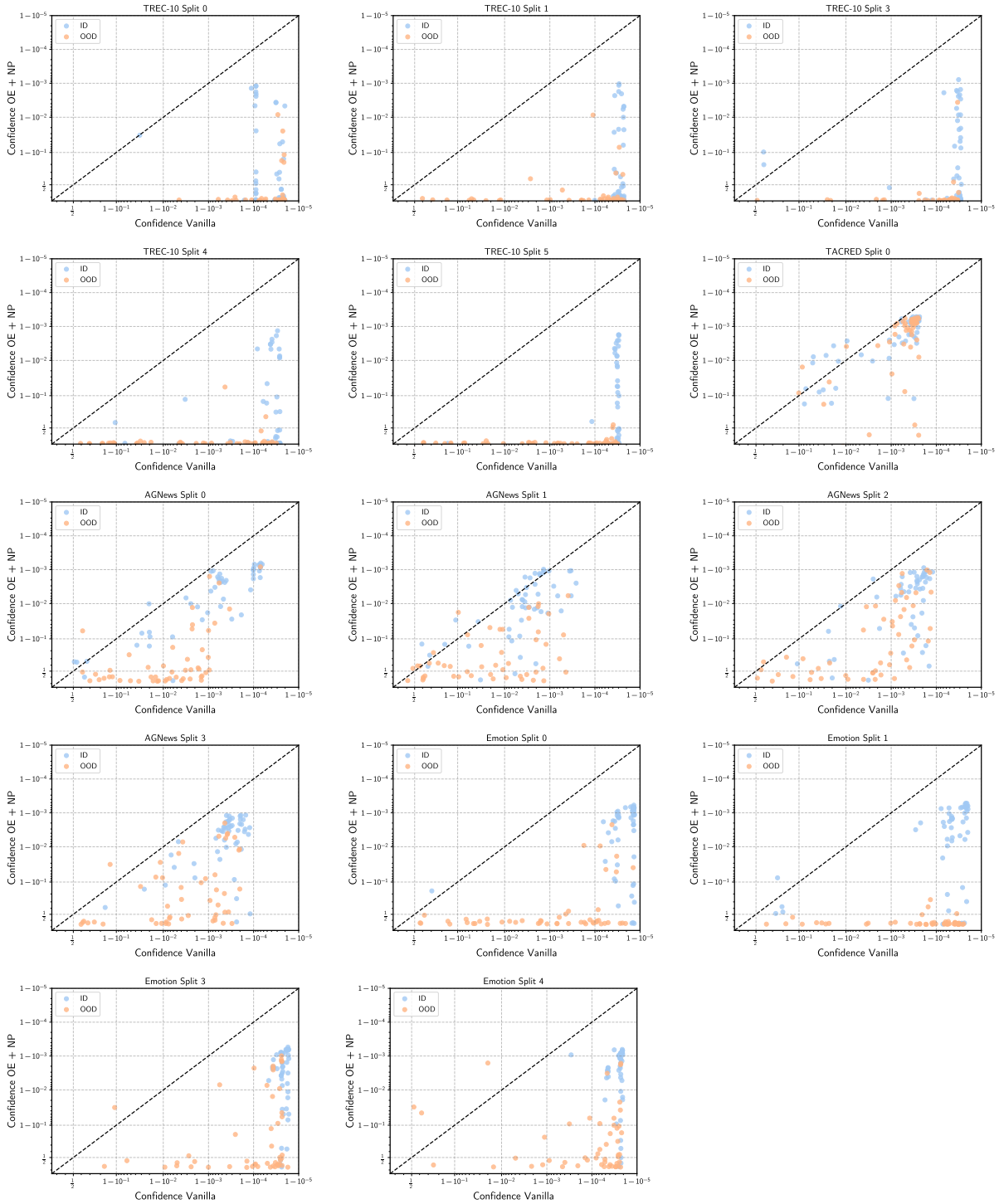

Figure 18: *OE decreases the confidence of OOD examples, but unfortunately also decreases confidence on ID examples.* We plot the confidences of 50 random ID and 50 random OOD examples on a vanilla finetuned BERT classifier versus a NP+OE trained BERT classifier. We also observe that ID examples exhibit a large confidence distribution after OE training: some ID examples have similar confidence as OOD examples. Note that the axis limits on these plots differ from the axis limits on Figure 17, as confidences in general are much lower.

(%)	$n =$	2	3	4	5	6	7
AGNews	Generation	61.6	23.4	8.8	4.1	2.2	1.3
	Heldout	70.4	36.7	20.4	13.1	9.0	6.7
TREC-10	Generation	46.2	21.4	10.3	4.1	1.7	0.7
	Heldout	47.5	23.9	11.1	5.1	2.6	1.4
Emotion	Generation	56.8	20.8	6.5	1.7	0.4	0.1
	Heldout	59.3	26.3	10.9	3.7	1.1	0.3
TACRED	Generation	86.8	68.1	57.2	47.9	40.2	32.9
	Heldout	87.2	72.3	63.6	58.2	54.2	50.5

Table 6: *Novel generations have lower n-gram overlap with the test set than heldout novel class examples.* We measure the percent of generation n-grams which appear in the test set, comparing this against a baseline of heldout novel class examples. Across all dataset and measured values of n, we find this overlap to be lower than baseline.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations. After section 7, on page 9

☑ A2. Did you discuss any potential risks of your work?
Limitations. After section 7, on page 9

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

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

B ☑ Did you use or create scientific artifacts?

Code described in Appendix A.2

☑ B1. Did you cite the creators of artifacts you used?
Section 4.1, Section 4.2

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Appendix A.2

☑ 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)?
Appendix A.2

☐ 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?

Section 4.2, Section 5, Appendix A.1, A.9, A.10, A.11

☑ 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 4.2, Appendix A.1

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 4.2, Appendix A.1, A.9, A.10, A.11

☑ 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?
Table 1. Table 2. Table 3. Table 5. Figure 6. Figure 12. We report the average over multiple seeds on all tables, and report standard error in subscript.

☐ 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.)?
Not applicable. Left blank.

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

Left blank.

☐ 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.?
Not applicable. Left blank.

☐ 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)?
Not applicable. Left blank.

☐ 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?
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.