Fine-Tuning Deteriorates General Textual Out-of-Distribution Detection by Distorting Task-Agnostic Features

Sishuo Chen[1], Wenkai Yang[1], Xiaohan Bi[1], Xu Sun[2]
[1]Center for Data Science, Peking University
[2]MOE Key Laboratory of Computational Linguistics, School of Computer Science, Peking University
{chensishuo,xusun}@pku.edu.cn
{wkyang,bxh}@stu.pku.edu.cn

Abstract

Detecting out-of-distribution (OOD) inputs is crucial for the safe deployment of natural language processing (NLP) models. Though existing methods, especially those based on the statistics in the feature space of fine-tuned pre-trained language models (PLMs), are claimed to be effective, their effectiveness on different types of distribution shifts remains underexplored. In this work, we take the first step to comprehensively evaluate the mainstream textual OOD detection methods for detecting semantic and non-semantic shifts. We find that: (1) no existing method behaves well in both settings; (2) fine-tuning PLMs on in-distribution data benefits detecting semantic shifts but severely deteriorates detecting non-semantic shifts, which can be attributed to the distortion of task-agnostic features. To alleviate the issue, we present a simple yet effective general OOD score named GNOME that integrates the confidence scores derived from the task-agnostic and task-specific representations. Experiments show that GNOME works well in both semantic and non-semantic shift scenarios, and further brings significant improvement on two cross-task benchmarks where both kinds of shifts simultaneously take place. Our code is available at https://github.com/lancopku/GNOME.

1 Introduction

The pre-training and fine-tuning paradigm based on Transformers (Vaswani et al., 2017) has achieved tremendous success in various natural language understanding (NLU) tasks (Devlin et al., 2019; Liu et al., 2019; Qiu et al., 2020). However, fine-tuned pre-trained language models (PLMs) notoriously suffer from over-confident predictions on out-of-distribution (OOD) inputs (Hendrycks et al., 2020). As this issue threats the reliability of NLP models deployed in the open world, textual OOD detection has attracted great attention recently (Podolskiy et al., 2021; Zhou et al., 2021, 2022; Duan et al., 2022, etc.), which aims to enable the model to abstain from making unreasonable predictions on OOD data and resort to human intervention.

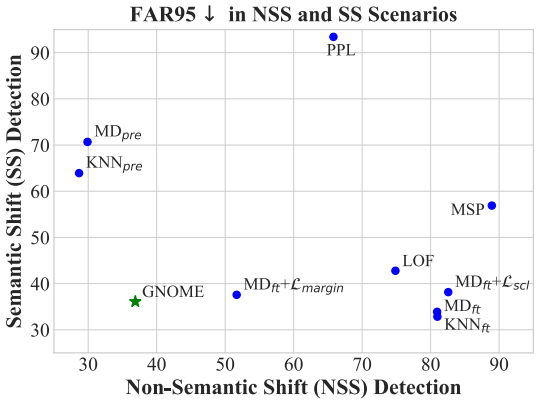


Figure 1: OOD detection performance (FAR95↓, lower is better) in non-semantic and semantic shift scenarios. No single existing method works well in both scenarios, but our proposed GNOME mitigates the trade-off.

Nonetheless, almost all of the current approaches are assessed under certain assumptions about the type of OOD texts. One line of works creates in-distribution (ID) and OOD pairs from arbitrary datasets for different tasks (Hendrycks et al., 2020), while another line assumes that OOD data belong to classes in the ID task but unseen during training, e.g., in intent recognition (Podolskiy et al., 2021). Arora et al. (2021) reveal the inconsistency among the evaluation protocols and category the distribution shifts to non-semantic shifts (NSS) and semantic shifts (SS), but a thorough comparison of existing methods in different settings is missing as later works either focus on either detecting NSS (Duan et al., 2022) or SS (Zhou et al., 2022).

In this work, we systematically evaluate the mainstream textual OOD detection methods on a comprehensive suite of benchmarks covering both NSS and SS scenarios. As shown in Figure 1, no single method wins across the board. Notably, the detec-

Findings of the Association for Computational Linguistics: EACL 2023, pages 564–579
May 2-6, 2023

tors based on the pre-trained features, e.g., the Mahalanobis distance detector MD_{pre} (Xu et al., 2021), excel at detecting non-semantic shifts but fail in detecting semantic shifts. In contrast, when the PLM is fine-tuned on annotated ID data, the detectors based on fine-tuned features, e.g., MD_{ft} (Podolskiy et al., 2021), perform well in the SS scenario but disastrously fail in the NSS setting. These observations uncover an intriguing trade-off: *fine-tuning contributes to the detection of semantic shifts but impairs the detection of non-semantic shifts*. This trade-off raises two critical research questions:

RQ1: *Why does fine-tuning undermine the detection of non-semantic shifts?* It is relatively easy to attribute the positive effect of fine-tuning in the SS setting to the learned class-discriminative features (Fort et al., 2021), but it remains unknown why fine-tuning plays a negative role in the NSS setting. We empirically find that the adverse effect comes from the fact that fine-tuning gradually destructs the pre-trained task-agnostic knowledge about general linguistic properties, which are useful cues for the detection of non-semantic shifts.

RQ2: *How to develop a general textual OOD detection method?* Since the type of distribution shifts is unknown in practice, our findings suggest that a practical method able to detect different kinds of OOD texts is yet to be developed. To this end, we aggregate the distance scores estimated in the feature space of both pre-trained and fine-tuned models to derive a **G**e**N**eral textual **O**OD **M**easurement scor**E** (GNOME) capable of detecting both NSS and SS. On the suite of benchmarks covering both NSS and SS settings, GNOME (the green star in Figure 1) surpasses the previous SOTA by 8.13 FAR95 points on average; on two cross-task benchmarks where both kinds of shifts happen simultaneously, GNOME reduces the average FAR95 by 4.88 points. Note that GNOME is not meant to be a SOTA method in all settings but rather a simple, principled way to get reasonable detection performance under various kinds of distribution shifts—we hope our analysis inspires better approaches for general textual OOD detection.

2 Related Work

OOD detection aims to detect abnormalities coming from a different distribution from the training data so that the model can refuse to make predictions on them (Amodei et al., 2016; Yang et al., 2021). Since it is essential for the security of machine learning models deployed in the open-world environment, OOD detection has gained great attention, first in computer vision (CV). We categorize the mainstream OOD detection methods into three groups by way to derive confidence scores: (1) *confidence-based methods* using the output probabilities of classifiers trained on in-distribution (ID) data (Hendrycks and Gimpel, 2017; Lakshminarayanan et al., 2017; Liang et al., 2018; Liu et al., 2020); (2) *density-based methods* using density scores derived from generation models (Zong et al., 2018; Ren et al., 2019; Xiao et al., 2020); (3) *distance-based methods* using the distance statistics in the feature space of neural networks (Lee et al., 2018; Huang et al., 2021; Sun et al., 2022).

Following the progress in CV, textual OOD detection based on PLMs has also attracted increasing attention. Hendrycks et al. (2020) show that the maximum softmax probability (MSP) score (Hendrycks and Gimpel, 2017) is a strong baseline for PLMs, followed by a group of works on confidence-based textual OOD detection (Li et al., 2021; Shen et al., 2021; Yilmaz and Toraman, 2022). As for the density-based branch, Gangal et al. (2020) and Arora et al. (2021) apply the idea to textual OOD detection by leveraging language models such as LSTM (Hochreiter and Schmidhuber, 1997) and GPT-2 (Radford et al., 2019). Regarding the distance-based methods, Podolskiy et al. (2021) revisit the Mahalanobis distance-based detector (Lee et al., 2018) for textual OOD detection based on fine-tuned PLMs and achieve performance gains over confidence-based methods, which is then further improved by introducing contrastive regularization (Zhou et al., 2021), utilizing nearest-neighbor distance (Zhou et al., 2022), and leveraging intermediate features (Chen et al., 2022).

Nonetheless, the NLP community lacks uniform evaluation criteria for OOD detection. Generally, ID/OOD pairs for evaluation are constructed in three ways: (1) *the non-semantic shift (NSS) setting* (a.k.a., the background shift setting) (Li et al., 2021; Arora et al., 2021; Duan et al., 2022), where ID and OOD data consist of the same semantic classes but differ in background information,[1] e.g.,

[1]Although the model can also make predictions on the samples with only non-semantic shifts, the accuracy tends to significantly drop. As the cost of wrong predictions is great in safety-critical scenarios, a conservative method for handling these samples by rejecting them is practical.

tweets as ID and Wikipedia comments as OOD in toxicity detection; (2) *the semantic shift (SS) setting* (Podolskiy et al., 2021; Zhou et al., 2022), where OOD data are composed of unseen classes belonging to the ID task, e.g., new classes in intent classification; (3) *the cross-task setting* (Hendrycks et al., 2020; Zhou et al., 2021), where the ID and OOD data are from datasets for different tasks and both semantic and non-semantic shifts happen, e.g., sentiment analysis data as ID and news classification data as OOD. Arora et al. (2021) first notice the inconsistency and compare confidence-based and density-based methods in NSS and SS settings, but they neglect the crucial branch of distance-based methods and the cross-task setting. In this work, we fill in this gap by presenting a comprehensive evaluation and developing a general textual OOD score motivated by our observations.

3 Observations and Explanations

In this section, we first give preliminaries in § 3.1. Then we introduce our benchmark for evaluating textual OOD detection (§ 3.2) and the evaluated methods (§ 3.3). Finally, we present the evaluation results (§ 3.4) and our interpretation of the observed trade-off between NSS and SS scenarios (§ 3.5).

3.1 Preliminaries

Problem Formulation The OOD detection problem can be formulated as a binary classification problem to decide whether an input example $\mathbf{x}$ belongs to the training data distribution $\mathcal{P}_{\text{in}}$ (ID) or not (OOD). An OOD detector D makes decisions for the input $\mathbf{x}$ based on the following formula:

$$D(\mathbf{x}) = \begin{cases} \text{ID} & \text{if } S(\mathbf{x}) \geq \gamma \\ \text{OOD} & \text{if } S(\mathbf{x}) < \gamma \end{cases}, \quad (1)$$

where $S(\mathbf{x})$ is the confidence score output by the detector and γ is the threshold chosen by the user.

Metrics We adopt two widely-used metrics AUROC and FAR95 following prior works (Podolskiy et al., 2021; Zhou et al., 2021). AUROC can be interpreted as the probability that the model ranks a random ID sample higher than a random OOD sample, and FAR95 is the proportion of negative samples (OOD) wrongly judged as positive (ID) when the true positive rate is 95%. Higher AUROCs and lower FAR95s indicate better performance.

Notations Assume M_θ is a PLM where θ denotes its parameters and $\mathbf{z} = M(\mathbf{x})$ denotes the feature vector for the input sample $\mathbf{x}$ derived from M (e.g., the last-layer CLS embedding in Transformers). For a classification task with C classes, the user fine-tunes M together with a classification head h and get the fine-tuned model $F_{\theta^*,h} = h \circ M_{\theta^*}$ where θ^* denotes the fine-tuned parameters. The output of F is $F_{\theta^*,h}(\mathbf{x}) = (p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_C(\mathbf{x}))^T$, which denotes the predicted probabilities.

Setting	Task	ID	OOD
Non-Semantic Shift	Sentiment Analysis	SST-2 IMDB	IMDB SST-2
	Toxic Detection	Twitter Jigsaw	Jigsaw Twitter
Semantic Shift	News Categorization	AGNews NC	$\text{AGNews}_{\text{OOD}}$ NC_{OOD}
	Dialogue Intent Classification	ROSTD CLINC	$\text{ROSTD}_{\text{OOD}}$ $\text{CLINC}_{\text{OOD}}$

Table 1: The architecture of the constructed suite of benchmarks categorized as either non-semantic shift (NSS) or semantic shift (SS). NC is short for the News Category dataset.

3.2 Benchmark Construction

We aim to build ID/OOD pairs where either the non-semantic shift (NSS) or the semantic shift (SS) dominates so that we can fairly compare existing methods on the ability to detect these two kinds of shifts separately. (1) For NSS, we choose SST-2 (Socher et al., 2013) and IMDB (Maas et al., 2011) for sentiment analysis, and Twitter (Founta et al., 2018) and Jigsaw[2] for toxicity detection. Among the four, any two datasets from the same task can be regarded as an ID/OOD pair. (2) For SS, we use four datasets: New Category (NC) (Misra and Grover, 2021; Misra, 2022), AGNews (Corso et al., 2005), ROSTD (Gangal et al., 2020), and CLINC (Larson et al., 2019). For each dataset, we use some classes as ID and the remaining classes as OOD. We show the architecture of the constructed suite of benchmarks categorized as either non-semantic shift (NSS) or semantic shift (SS) in Table 1 and more details can be found in Appendix A. Compared with Arora et al. (2021), we additionally include toxicity detection data for NSS and intent recognition data for SS, which make the suite of benchmarks more representative of real-world scenarios.

[2]Available at this link.

3.3 Evaluated Baselines

OOD Detection Methods We evaluate the mainstream methods as follows. (1) For *confidence-based methods*, we test the MSP baseline ($S(\mathbf{x}) = \max_{y\in\{1,2,\ldots,C\}} p_y(\mathbf{x})$) (Hendrycks and Gimpel, 2017) and its three variants: Scaling (Liang et al., 2018), Energy Score (Liu et al., 2020), and D2U (Yilmaz and Toraman, 2022); (2) For *density-based methods*, we evaluate the PPL method (Arora et al., 2021) using the GPT-2 model for language modeling ($S(\mathbf{x}) = 1/\mathrm{PPL}(\mathbf{x})$); (3) For *distance-based methods*, we test the LOF method (Lin and Xu, 2019) that trains a local outlier detector on fine-tuned features of ID data, and the basic variants of the Mahalanobis detector (MD): $\mathrm{MD}_{\mathrm{pre}}$ (Xu et al., 2021) built on pre-trained features ($\mathbf{z} = M_\theta(\mathbf{x})$) and $\mathrm{MD}_{\mathrm{ft}}$ (Podolskiy et al., 2021) built on fine-tuned features ($\mathbf{z} = M_{\theta^*}(\mathbf{x})$). Also, we evaluate two variants of MD built on features derived from PLMs fine-tuned with supervised contrastive and margin-based auxiliary targets (Zhou et al., 2021), namely $\mathrm{MD}_{\mathrm{ft}} + \mathcal{L}_{\mathrm{scl}}$ and $\mathrm{MD}_{\mathrm{ft}} + \mathcal{L}_{\mathrm{margin}}$. Generally, the confidence score in MD is formulated as:

$$\begin{aligned} \mathrm{MD}(\mathbf{x}) &= \min_{c\in\{1,2,\ldots,C\}} (\mathbf{z}-\mu_c)^T \Sigma^{-1} (\mathbf{z}-\mu_c), \\ S(\mathbf{x}) &= -\mathrm{MD}(\mathbf{x}), \end{aligned} \tag{2}$$

where μ_c is the class centroid for class c and Σ is the global covariance matrix (μ and Σ can be estimated on ID training data). Besides, we evaluate the nearest-neighbor detectors (Sun et al., 2022) based on pre-trained ($\mathrm{KNN}_{\mathrm{pre}}$) and fine-tuned features ($\mathrm{KNN}_{\mathrm{ft}}$). We refer readers to Appendix C for more details about the baselines.

Model Configuration For the methods based on fine-tuned PLMs, we build text classifiers by fine-tuning the $\mathrm{RoBERTa}_{\mathrm{base}}$ (Liu et al., 2019) model (110M parameters) on annotated ID data. For $\mathrm{MD}_{\mathrm{pre}}$, we use the pre-trained $\mathrm{RoBERTa}_{\mathrm{base}}$ model. For the PPL method, we fine-tune the $\mathrm{GPT\text{-}2}_{\mathrm{small}}$ model (117M parameters) for language modeling on ID data. More details can be found in Appendix B.

3.4 Evaluation Results and Findings

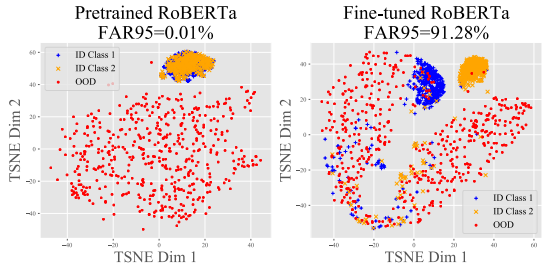
We display the main evaluation results in Table 2. As shown, the confidence-based methods underperform the density-based method PPL in the NSS setting, while they outrival PPL in the SS setting, in line with the observations in Arora et al. (2021). Notably, we notice that the distance-based methods achieve the best results in both NSS and SS settings. Concretely, $\mathrm{MD}_{\mathrm{pre}}$ and $\mathrm{KNN}_{\mathrm{pre}}$ built on pre-trained features are the best in the NSS setting, while $\mathrm{MD}_{\mathrm{ft}}$ and $\mathrm{KNN}_{\mathrm{ft}}$ built on fine-tuned features are the best in the SS setting. However, no single method wins across the board. Thus, we draw an intriguing trade-off: *In textual OOD detection, fine-tuning PLMs on ID data boosts semantic shift detection but impairs non-semantic shift detection.*

Category	Method	Avg.	NSS	SS
Confidence	MSP	71.63/72.92	65.47/88.94	77.78/56.89
	Scaling	71.96/71.62	65.45/88.94	78.47/54.30
	Energy	71.75/71.63	64.90/89.05	78.61/54.20
	D2U	71.99/71.49	65.47/88.94	78.52/54.04
Density	PPL	67.65/79.61	74.28/65.81	61.03/93.42
Distance	$\mathrm{MD}_{\mathrm{ft}}$	80.39/57.42	72.25/80.95	88.54/33.89
	$\mathrm{MD}_{\mathrm{ft}} + \mathcal{L}_{\mathrm{scl}}$	82.22/60.36	76.71/82.57	87.73/38.15
	$\mathrm{MD}_{\mathrm{ft}} + \mathcal{L}_{\mathrm{margin}}$	**86.50/44.63**	85.45/51.68	87.54/37.57
	$\mathrm{MD}_{\mathrm{pre}}$	83.76/50.29	**93.29**/29.90	74.22/70.68
	$\mathrm{KNN}_{\mathrm{ft}}$	81.02/56.91	72.58/80.99	**89.47/32.84**
	$\mathrm{KNN}_{\mathrm{pre}}$	85.69/46.29	92.66/**28.67**	78.72/63.92

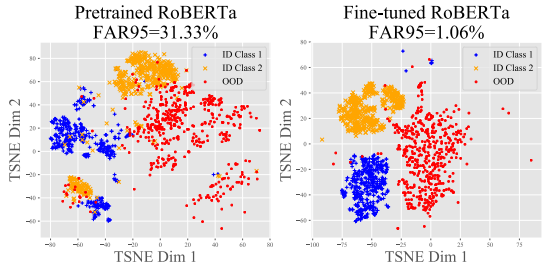
Table 2: The performance (AUROC↑/FAR95↓ values in percentage) of the evaluated approaches. All results are averaged over five random seeds, and best results are highlighted in **bold**. We report results averaged on the ID/OOD pairs in NSS and SS setting in the last two columns, respectively, and report the results averaged on all eight benchmarks in the third column. See full results on each benchmark in Table 4 and Appendix E.

To intuitively understand the effect of fine-tuning, we visualize the features using t-SNE (Van der Maaten and Hinton, 2008) in both settings. As plotted in Figure 2, before fine-tuning, the ID and OOD samples are sharply separated in the NSS setting, but show a significant overlap in the SS setting; after fine-tuning, the ID samples are well clustered on class in both settings and unseen classes (OOD) in the SS setting are also pulled away from the ID data, but OOD samples in the NSS setting become almost indistinguishable from the ID data.

The observed benefits of fine-tuning in the SS setting match the observation in the near-OOD image detection (Fort et al., 2021), suggesting that the fine-tuned task-specific representations are more suitable for detecting unseen classes belonging to the ID task. Regarding the negative effect of fine-tuning in the NSS setting, we speculate that it can be explained in this way: task-agnostic features important for detecting non-semantic shifts are learned during pre-training but discarded in the fine-tuning stage, for which we will present empirical evidence in § 3.5. To our knowledge, we are the first to study the impact of fine-tuning on the de-



(a) Non-Semantic Shift: SST-2 (ID) vs. IMDB (OOD)



(b) Semantic Shift: ROSTD (ID) vs. $\text{ROSTD}_{\text{OOD}}$ (OOD)

Figure 2: T-SNE visualizations for the features derived from pre-trained and fine-tuned RoBERTa models and the corresponding FAR95 of the Mahalanobis detector.

tection of different kinds of OOD texts and reveal the trade-off between the NSS setting (fine-tuning harms) and the SS setting (fine-tuning helps).

In addition, we notice that when the model is fine-tuned with margin-based contrastive auxiliary targets ($\mathcal{L}_{\text{margin}}$) (Zhou et al., 2021), the MD detector (MD_{ft}+$\mathcal{L}_{\text{margin}}$) substantially surpasses MD_{ft} in the NSS setting with marginal sacrifice in the SS setting, thus it achieves the best performance on average. However, it still falls far behind MD_{pre} in the NSS setting. *As no single existing method behaves well in both settings, a general textual OOD detection method capable of detecting different kinds of OOD texts is yet to be developed, given the broad range of distribution shifts in realistic scenarios.*

3.5 Empirical Explanations

Probing Analysis. From the view of the oracle, the OOD data in the NSS setting can be easily distinguished from the ID data by certain task-agnostic linguistic features. For example, the IMDB data are long movie reviews with an average length of 230 tokens, which can be well distinguished by length from the SST-2 reviews with an average length of 19 tokens. Therefore, we speculate that the negative effect of fine-tuning arises from the deletion of general linguistic features during fine-tuning. To test the conjecture, we evaluate the sentence em-

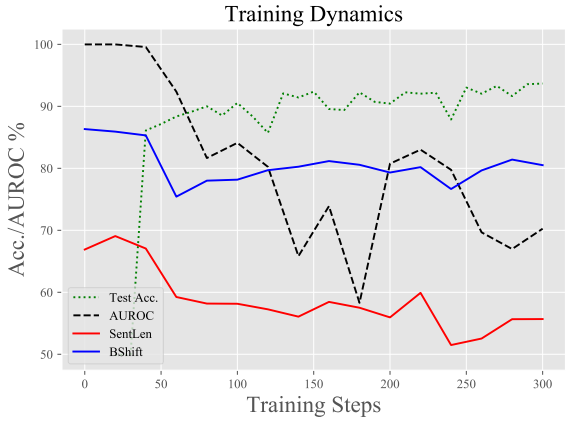


Figure 3: The dynamics of test accuracy, probing accuracies of *SentLen* and *BShift*, and the performance (AUROC↑) to detect IMDB as OOD in the fine-tuning process of the RoBERTa model on SST-2. We have seen similar trends on other datasets in the NSS setting.

Methods	Avg.	NSS	SS
MD_{pre}	50.29	**29.90**	70.68
MD_{ft}	57.42	80.95	33.89
MD_{ft} + RecAdam	55.76	78.39	33.13
MD_{ft} (head lr × 10)	53.90	76.18	**31.63**
MD_{ft} + LP-FT	50.92	60.26	41.59

Table 3: The performance of MD_{ft} coupled with regularization techniques on textual OOD detection. We report the FAR95↓ values on average.

beddings produced by the model checkpoints in the fine-tuning process on SST-2 on two classic probing tasks designed by Conneau and Kiela (2018): *SentLen* (sentence length) and *BShift* (bigram shift). They are general linguistic features irrelevant to the class labels in downstream classification tasks, so the probing accuracies can be regarded as indicators of the preservation of task-agnostic features (see details in Appendix A.3). We show the tendency of the probing accuracies along with corresponding OOD detection performance (IMDB as OOD) and test accuracy in Figure 3. We find that as fine-tuning goes on, although the classification performance on ID test data shows an upward trend, the OOD detection performance (AUROC) gradually declines along with the probing accuracies. The observed correlations between the OOD detection performance and the probing accuracies on *SentLen* and *BShift* empirically indicate that fine-tuning impairs NSS detection by distorting the task-agnostic features in pre-trained models.

Does Regularization Help? As stated, fine-tuning may destruct the pre-trained features and

thus harm NSS detection. If the cause-effect holds, the negative effect can be alleviated by regularization techniques for preserving pre-trained features. To verify this deduction, we investigate three regularization approaches: (1) the RecAdam optimizer (Chen et al., 2020); (2) a 10× larger learning rate for the head (Prabhu et al., 2021); (3) the linear-probing then fine-tuning approach (LP-FT) (Kumar et al., 2022). As shown in Table 3, the regularization techniques applied to MD_{ft} bring moderate improvements in the NSS setting, but they still fall far behind MD_{pre} that exploits the original pre-trained features. The results provide further empirical support for our reasoning about the demerits of fine-tuning. Moreover, they suggest that effectively preserving pre-trained task-agnostic features suitable for NSS detection in fine-tuned PLMs is challenging. A plausible solution is to decouple task-agnostic and task-specific features in a single fine-tuned model, which we leave for future study. Another possible solution is to directly leverage the pre-trained model, which we will introduce next.

4 GNOME for Textual OOD Detection

In view of the observed trade-off, we are motivated to combine the strengths of task-agnostic and task-specific representations to obtain a confidence score capable of modeling both non-semantic shifts and semantic shifts. A straightforward way is to take the mean of MD_{pre} and MD_{ft} scores.[3] However, given that the norm of features can fluctuate and thus the distance scores are not comparable across different spaces, simple averaging may cause the integrated score to be skewed towards the side with the larger norm. To alleviate the issue, we normalize MD_{pre} and MD_{ft} before aggregation:

$$\begin{aligned} \text{Norm}(\text{MD}_{\text{pre}}(\mathbf{x})) &= \frac{\text{MD}_{\text{pre}}(\mathbf{x}) - \mu_{\text{pre}}}{\sigma_{\text{pre}}}, \\ \text{Norm}(\text{MD}_{\text{ft}}(\mathbf{x})) &= \frac{\text{MD}_{\text{ft}}(\mathbf{x}) - \mu_{\text{ft}}}{\sigma_{\text{ft}}}, \end{aligned} \tag{3}$$

where μ and σ are the mean and standard deviation of Mahalanobis distance scores, respectively, which can be estimated on ID validation samples. Then we obtain the integrated score GNOME (**G**e**N**eral textual **O**OD **M**easurement scor**E**):

$$S_{\text{GNOME}}(\mathbf{x}) = -\text{Agg}(\text{Norm}(\text{MD}_{\text{pre}}(\mathbf{x})), \text{Norm}(\text{MD}_{\text{ft}}(\mathbf{x})), \tag{4}$$

where Agg is the aggregation operator (such as the mean or max). We use the mean operator for aggregation in our main experiments. Note that We do not use a weighted average because it is not possible to tune the weights when OOD data is unknown, which follows the mainstream setting in OOD detection. If the user has prior knowledge about the type of OOD data, he/she can train the weights to aggregate the scores.

5 Experiments

5.1 Experimental Setup

Benchmarks Besides the eight benchmarks introduced in § 3.2 that are categorized as either NSS or SS, we also evaluate GNOME and baselines in the cross-task setting (Hendrycks et al., 2020; Zhou et al., 2021) where both kinds of shifts happen simultaneously. Following Zhou et al. (2021), we choose SST-2 (Socher et al., 2013) and 20 Newsgroups (Lang, 1995) as ID data, and regard a series of datasets from different tasks as OOD data: TREC-10 (Li and Roth, 2002), WMT-16 (Bojar et al., 2016), Multi30k (Elliott et al., 2016), RTE (Dagan et al., 2005), and SNLI (Bowman et al., 2015). Refer to Appendix A for more details.

Models and Metrics We follow the same model configuration as that in § 3.3 in main experiments and report FAR95 values in the main text (the trend of AUROC results in Appendix E is similar).

5.2 Results and Analysis

GNOME works well in both SS and NSS settings and significantly surpasses baselines in terms of average performance. As shown in Table 4, GNOME is competent in both settings (close to MD_{pre} for NSS and MD_{ft} for SS) and achieves the best performance on average. The average FAR95 is 36.50%, 8.13% lower than the previous SOTA MD_{ft}+$\mathcal{L}_{\text{margin}}$ requiring extra margin-based targets.

GNOME also achieves superior performance in the cross-task setting. As results in Table 5, GNOME outperforms all baseline methods (4.88% FAR95 reduction on average) in the cross-task setting, demonstrating the power of integrating task-agnostic and task-specific representations when non-semantic shifts and semantic shifts happen simultaneously. Note that among existing methods, MD_{pre} and KNN_{pre} are the best on the 20 Newsgroups benchmark, suggesting that non-semantic shifts dominate there; MD_{ft}+$\mathcal{L}_{\text{margin}}$ is the best on

[3] Our core idea is orthogonal to the distance-based scoring function, so we can also combine KNN_{pre} and KNN_{ft}. We have tested in this way and got similar results to those obtained by combining MD_{pre} and MD_{ft}.

Methods	Avg.	Non-Semantic Shift (NSS)				Semantic Shift (SS)			
		SST-2	IMDB	Twitter	Jigsaw	NC	AGNews	ROSTD	CLINC
MSP (Hendrycks and Gimpel, 2017)	72.92	90.50	79.20	89.76	96.29	72.16	85.06	51.24	19.08
Scaling (Liang et al., 2018)	71.62	90.50	79.20	89.76	96.29	68.94	80.19	52.15	15.90
Energy (Liu et al., 2020)	71.63	90.58	79.65	89.70	96.27	69.35	79.17	52.53	15.76
D2U (Yilmaz and Toraman, 2022)	71.49	90.50	79.20	89.76	96.29	68.86	79.49	52.14	15.66
PPL (Arora et al., 2021)	79.61	64.65	7.25	94.53	96.79	93.50	95.78	86.99	97.40
LOF (Lin and Xu, 2019)	58.82	93.39	24.28	88.86	92.90	70.83	81.00	4.69	14.58
MD_{ft} (Podolskiy et al., 2021)	57.42	91.28	47.78	88.49	96.25	58.16	64.06	1.06	**12.26**
MD_{ft} + $\mathcal{L}_{scl}$ (Zhou et al., 2021)	60.36	88.05	63.11	84.53	94.57	67.00	69.26	3.42	12.90
MD_{ft} + $\mathcal{L}_{margin}$ (Zhou et al., 2021)	44.63	32.61	4.70	72.51	96.90	59.31	67.48	11.07	12.40
MD_{pre} (Xu et al., 2021)	50.29	0.01	1.54	44.40	**73.65**	90.14	87.64	31.33	73.30
KNN_{ft} (Sun et al., 2022)	56.91	87.87	56.81	83.67	95.60	**56.51**	**60.64**	**0.71**	13.50
KNN_{pre} (Sun et al., 2022)	46.29	**0.00**	**1.48**	**33.11**	80.03	84.56	81.89	18.51	70.70
GNOME (Ours)	**36.50**	0.04	8.24	53.36	85.88	64.81	63.25	1.47	14.94

Table 4: OOD detection performance (FAR95↓, lower is better) on the constructed suite of benchmarks. All values are percentages averaged over five different random seeds, and the best results are highlighted in **bold**. The second column gives the average performance on eight benchmarks.

Methods	Avg.	ID Datasets	
		SST-2	20 NG
MSP	59.98	70.00	49.95
Scaling	50.68	70.00	31.36
Energy	52.31	72.43	32.31
D2U	51.15	70.00	32.29
LOF	51.55	66.29	36.81
MD_{ft}	32.29	48.82	15.75
MD_{ft} + $\mathcal{L}_{scl}$	35.30	49.04	21.56
MD_{ft} + $\mathcal{L}_{margin}$	23.97	29.43	18.51
MD_{pre}	17.90	35.79	**0.01**
KNN_{ft}	43.58	63.73	23.42
KNN_{pre}	20.79	41.57	**0.01**
GNOME (ours)	**13.02**	**26.02**	**0.01**

Table 5: OOD detection performance in the cross-task setting. For each ID dataset, we report the macro average of FAR95↓ on all corresponding OOD datasets, averaged over five random seeds.

the SST-2 benchmark, indicating that both kinds of shifts matter there. Without any prior knowledge about the type of distribution shifts, GNOME yields the best performance on both benchmarks.

5.3 Ablation Study

We examine the rationality of the key components of GNOME here. As shown in Table 6, when the normalization operation is absent, the performance in the SS setting is slightly enhanced (~3% FAR95 reduction), but the performance in the NSS and cross-task settings drops by around 7% FAR95 points, which suggests that the normalization operation helps strike a balance between the two scenarios and thus achieve better performance on average. These results also empirically verify that the mean operator is more suitable than the max operator for the score aggregation step in GNOME. We have also tested other common normalization methods such as min-max and found that they underperform the standardization normalization employed in GNOME.

Norm.	Agg.	Avg.	NSS	SS	CT
✓	mean	**28.67**	**36.88**	36.12	**13.02**
	max	30.08	38.33	37.90	14.01
✗	mean	32.61	44.00	**33.32**	20.52
	max	34.67	45.31	33.88	24.83

Table 6: The performance (FAR95↓) corresponding to different normalization choices and score aggregators in GNOME. CT denotes the cross-task setting.

Besides the score-level fusion in GNOME, we have also tested feature-level fusion (concatenating or averaging pre-trained and fine-tuning features), but they lead to a significant drop in the SS setting (+20% FAR95) while only a slight improvement in the SS setting. Thus we argue that the score-level fusion by the mean operator is better.

6 Further Discussion

6.1 Comparison with Ensemble Methods

On the top of MD_{ft} based on the fine-tuned PLM, GNOME is free of modification to the model architecture or training, and only requires an extra inference of the off-shelf PLM to obtain pre-trained features, thus being practical for real-world deployment. For a strictly fair comparison under the same inference overhead constraint, we compare GNOME with previous ensemble meth-

Methods	#Passes	Avg.	NSS	SS
Single Pass				
MSP	1	72.92	88.94	56.89
MD_{ft}	1	57.42	80.95	33.89
MD_{pre}	1	50.29	**29.90**	70.68
Model Ensemble				
MSP	2	70.25	88.45	52.05
MD_{ft}	2	56.09	81.35	30.82
MSP	5	68.59	88.37	48.80
MD_{ft}	5	55.35	80.64	**30.11**
Dropout Ensemble				
MC Dropout	2	72.68	88.41	56.95
MC Dropout	5	70.75	85.88	55.62
GNOME	2	**36.50**	36.88	36.10

Table 7: Comparison with ensemble methods on the developed benchmark. We report FAR95 values averaged on the ID/OOD pairs in both SS and NSS settings.

ods, which can be divided into two groups: (1) *Model ensemble* (Lakshminarayanan et al., 2017): summing confidence scores derived from models trained over different random seeds (we apply it to MSP and MD_{ft}); (2) *MC Dropout* (Gal and Ghahramani, 2016): summing the probabilities output by multiple inferences with dropout on.

As the results shown in Table 7, previous ensemble methods that require 2× or 5× forward passes only slightly raise the performance compared with their single-pass counterparts, and fall far behind GNOME in terms of the average detection performance. These results also substantiate the power of integrating pre-trained and fine-tuned features. We do not compare with the k-Folden method (Li et al., 2021) that needs $(C-1)$ sub-models (C is the number of ID classes) because it does not apply to binary classification problems and is expensive for large-scale problems where C is large.

6.2 The Choice of Pre-Trained Features

In the main experiments, we adopt the last-layer CLS embeddings as the pre-trained features for simplicity and fair comparison between MD_{pre} and MD_{ft}. As works on unsupervised textual OOD detection (Xu et al., 2021) and unsupervised sentence embedding (Su et al., 2021) show, pooling operations such as token-level and layer-level averaging produce better pre-trained features. We then alternatively use *last-avg* (the average of token embeddings in the last layer) and *first-last-avg* (the average of token embeddings in the first and last layers) embeddings as pre-trained features in MD_{pre} and GNOME. As shown in Table 8, when

Pre-trained Features	Methods	Avg.	NSS	SS
-	MD_{ft}	57.42	80.95	33.89
last-cls	MD_{pre}	50.29	**29.90**	70.68
	GNOME	36.50	36.88	36.10
last-avg	MD_{pre}	46.28	36.14	56.41
	GNOME	**35.88**	38.33	**33.43**
first-last-avg	MD_{pre}	41.77	36.00	47.54
	GNOME	35.89	37.93	33.85

Table 8: The OOD detection performance (FAR95↓ in percentage) of different pre-trained features.

Backbone	Methods	Avg.	NSS	SS
$BERT_{base\text{-}uncased}$	MD_{pre}	67.81	60.61	75.01
	MD_{ft}	59.06	85.05	33.08
	GNOME	50.98	65.42	36.55
$RoBERTa_{base}$	MD_{pre}	50.29	29.90	70.68
	MD_{ft}	57.42	80.95	33.89
	GNOME	36.50	36.88	36.10
$RoBERTa_{large}$	MD_{pre}	71.17	58.72	83.60
	MD_{ft}	58.98	83.33	34.62
	GNOME	44.92	54.08	35.76

Table 9: Textual OOD detection performance (FAR95↓ values on average) with different pre-trained backbones.

the *last-cls* embeddings are replaced with the *last-avg* or *first-last-avg* embeddings, MD_{pre} is moderately degraded in the NSS setting (~7% FAR95 increase), but it is drastically improved in the SS setting (~14% or ~23% FAR95 reduction). Notably, the trade-off before and after fine-tuning still holds when the *last-avg* or *first-last-avg* is used to get pre-trained features. However the pre-trained features are derived, GNOME consistently brings improvements to the average detection performance.

6.3 Generalization on Other PLMs

To demonstrate the generality of GNOME, we also test on another two PLMs: $BERT_{base\text{-}uncased}$ (Devlin et al., 2019) (110M parameters) and $RoBERTa_{large}$ (Liu et al., 2019) (355M parameters). As shown in Table 9, we observe that: (1) The NSS-SS trade-off is prevalent on different PLMs and GNOME brings consistent gains over baselines in terms of average performance. (2) $RoBERTa_{base}$, which uses more diverse pre-training data, beats $BERT_{base\text{-}uncased}$, suggesting that pre-training on diverse data boosts textual OOD detection; $RoBERTa_{large}$ underperforms $RoBERTa_{base}$, indicating that larger models are not necessarily better at OOD detection.

7 Conclusion

Aware of the lack of a fair and comprehensive evaluation of current textual OOD detection methods, we take the first step to systematically assess them under different distribution shifts. Interestingly, we find that no single method works well in both the non-semantic shift setting and the semantic shift setting, and there exists a trade-off: fine-tuning pre-trained language models on in-distribution data benefits detecting semantic shifts but undermines detecting non-semantic shifts. After presenting empirical explanations for the trade-off from the perspective of feature distortion, we are then motivated to fully utilize both the pre-trained and fine-tuned features to obtain an efficient measurement score GNOME for better detecting diverse distribution shifts. Extensive experimental results demonstrate the efficacy and generality of GNOME. Overall, GNOME is a first step in leveraging the intuition from our observations and analysis, and we hope that this work sheds light on the behavior of pre-trained language models upon detecting different kinds of distribution shifts and inspires new methods for general textual OOD detection.

Limitations

Although our approach GNOME yields the best overall performance on the suite of benchmarks where either NSS or SS dominates and also performs best in the cross-task setting where both kinds of shifts take place, it slightly underperforms MD_{pre} and KNN_{pre} in the NSS setting and marginally lags behind MD_{ft} and KNN_{ft} in the SS setting. This is comprehensible because it is challenging for a single method to function perfectly for arbitrary OOD data without priors on the type of distribution shifts as analyzed in visual OOD detection works (Ahmed and Courville, 2020). Note again that we do not intend to present a perfect textual OOD detector capable of tackling all kinds of distribution shifts; instead, our core contributions are that we discover the trade-off between NSS and SS settings, present an empirical analysis to explain the phenomenon and provide insights to mitigate the trade-off for general textual OOD detection.

Ethical Considerations

We believe that our work leads to a better understanding of the behavior of pre-trained language models on OOD texts. We also believe that the proposed method will facilitate the reliable deployment of NLU models since a model may face various types of OOD inputs in the wild and our method contributes to the detection performance on unknown OOD data in the average sense. All experiments in this work are conducted on open datasets and all pre-trained models that we investigate are publicly available. We do not anticipate any negative social consequences to our work and we hope to continue to build on our method and develop more effective textual OOD detectors in the future.

Acknowledgement

We sincerely thank all the anonymous reviewers for their valuable comments and advice. This work is supported by Natural Science Foundation of China (NSFC) No. 62176002. Xu Sun is the corresponding author of this paper.

References

Faruk Ahmed and Aaron C. Courville. 2020. Detecting semantic anomalies. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 3154–3162. AAAI Press.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. 2016. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*.

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 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 10687–10701. Association for Computational Linguistics.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, et al. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 131–198.

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

Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. 2000. Lof: identifying density-

based local outliers. In *ACM sigmod record*, volume 29, pages 93–104. ACM.

Sanyuan Chen, Yutai Hou, Yiming Cui, Wanxiang Che, Ting Liu, and Xiangzhan Yu. 2020. Recall and learn: Fine-tuning deep pretrained language models with less forgetting. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Sishuo Chen, Wenkai Yang, Xiaohan Bi, and Xu Sun. 2022. Holistic sentence embeddings for better out-of-distribution detection. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6676–6686, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single \$&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2126–2136. Association for Computational Linguistics.

Gianna M. Del Corso, Antonio Gulli, and Francesco Romani. 2005. Ranking a stream of news. In *Proceedings of the 14th international conference on World Wide Web, WWW 2005, Chiba, Japan, May 10-14, 2005*, pages 97–106. ACM.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, pages 177–190. Springer.

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.

Hanyu Duan, Yi Yang, Ahmed Abbasi, and Kar Yan Tam. 2022. Barle: Background-aware representation learning for background shift out-of-distribution detection. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 750–764, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. 2016. Multi30k: Multilingual english-german image descriptions. *arXiv preprint arXiv:1605.00459*.

Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. 2021. Exploring the limits of out-of-distribution detection. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 7068–7081.

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

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1050–1059. JMLR.org.

Varun Gangal, Abhinav Arora, Arash Einolghozati, and Sonal Gupta. 2020. Likelihood ratios and generative classifiers for unsupervised out-of-domain detection in task oriented dialog. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7764–7771. AAAI Press.

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 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2744–2751, Online. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. 2020. Generalized ODIN: detecting out-of-distribution image without learning from out-of-distribution data. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10948–10957. IEEE.

Haiwen Huang, Zhihan Li, Lulu Wang, Sishuo Chen, Xinyu Zhou, and Bin Dong. 2021. Feature space singularity for out-of-distribution detection. In *Proceedings*

of the Workshop on Artificial Intelligence Safety 2021 (SafeAI 2021) co-located with the Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI 2021), Virtual, February 8, 2021, volume 2808 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

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

Ananya Kumar, Aditi Raghunathan, Robbie Matthew Jones, Tengyu Ma, and Percy Liang. 2022. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*.

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 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6402–6413.

Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Machine Learning, Proceedings of the Twelfth International Conference on Machine Learning, Tahoe City, California, USA, July 9-12, 1995*, pages 331–339. Morgan Kaufmann.

Stefan Larson, Anish Mahendran, Joseph J Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K Kummerfeld, Kevin Leach, Michael A Laurenzano, Lingjia Tang, et al. 2019. An evaluation dataset for intent classification and out-of-scope prediction. *arXiv preprint arXiv:1909.02027*.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 7167–7177.

Xiaoya Li, Jiwei Li, Xiaofei Sun, Chun Fan, Tianwei Zhang, Fei Wu, Yuxian Meng, and Jun Zhang. 2021. kfolden: k-fold ensemble for out-of-distribution detection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 3102–3115. Association for Computational Linguistics.

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

Shiyu Liang, Yixuan Li, and R. Srikant. 2018. Enhancing the reliability of out-of-distribution image detection in neural networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

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*, pages 5491–5496, Florence, Italy. Association for Computational Linguistics.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. 2020. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems*, volume 33, pages 21464–21475. Curran Associates, Inc.

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. *arXiv preprint arXiv:1907.11692*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 142–150. The Association for Computer Linguistics.

Rishabh Misra. 2022. News category dataset. *arXiv preprint arXiv:2209.11429*.

Rishabh Misra and Jigyasa Grover. 2021. *Sculpting Data for ML: The first act of Machine Learning*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Alexander Podolskiy, Dmitry Lipin, Andrey Bout, Ekaterina Artemova, and Irina Piontkovskaya. 2021. Revisiting mahalanobis distance for transformer-based out-of-domain detection. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13675–13682. AAAI Press.

Viraj Prabhu, Shivam Khare, Deeksha Kartik, and Judy Hoffman. 2021. Sentry: Selective entropy optimization via committee consistency for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF*

International Conference on Computer Vision, pages 8558–8567.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Jie Ren, Peter J. Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark A. DePristo, Joshua V. Dillon, and Balaji Lakshminarayanan. 2019. Likelihood ratios for out-of-distribution detection. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 14680–14691.

Yilin Shen, Yen-Chang Hsu, Avik Ray, and Hongxia Jin. 2021. Enhancing the generalization for intent classification and out-of-domain detection in SLU. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2443–2453. Association for Computational Linguistics.

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

Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

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

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 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.

Zhisheng Xiao, Qing Yan, and Yali Amit. 2020. Likelihood regret: An out-of-distribution detection score for variational auto-encoder. *Advances in Neural Information Processing Systems*, 33.

Keyang Xu, Tongzheng Ren, Shikun Zhang, Yihao Feng, and Caiming Xiong. 2021. Unsupervised out-of-domain detection via pre-trained transformers. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 1052–1061. Association for Computational Linguistics.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. 2021. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*.

Eyup Yilmaz and Cagri Toraman. 2022. D2U: Distance-to-uniform learning for out-of-scope detection. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2093–2108, Seattle, United States. Association for Computational Linguistics.

Wenxuan Zhou, Fangyu Liu, and Muhao Chen. 2021. Contrastive out-of-distribution detection for pretrained transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 1100–1111. Association for Computational Linguistics.

Yunhua Zhou, Peiju Liu, and Xipeng Qiu. 2022. KNN-contrastive learning for out-of-domain intent classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5129–5141, Dublin, Ireland. Association for Computational Linguistics.

Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Dae-ki Cho, and Haifeng Chen. 2018. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

A Dataset Introduction and Statistics

A.1 The Constructed Suite of Benchmarks

We show the included datasets in Table 10 and give an introduction to them as follows.

For the NSS setting, we consider two tasks: sentiment analysis and toxic detection. For sentiment analysis, we choose SST-2 (Socher et al., 2013) and IMDB (Maas et al., 2011). SST-2 contains short

Dataset	# Classes	# Train	# Dev	# Test	L
SST-2	2	6,920	872	1,821	19
IMDB	2	23,000	2,000	25,000	230
Twitter	2	69,632	7,737	8,597	17
Jigsaw	2	143,614	15,957	63,978	68
AGNews	4	115,778	3,994	3,993	23
NC	5	68,859	8,617	8,684	30
ROSTD	12	30,521	4,181	8,621	7
CLINC	150	15,000	3,000	4,500	8
$\text{AGNews}_{\text{OOD}}$	-	-	-	3,600	21
NC_{OOD}	-	-	-	11,402	29
$\text{ROSTD}_{\text{OOD}}$	-	-	-	3,090	7
$\text{CLINC}_{\text{OOD}}$	-	-	-	1,000	9

Table 10: Statistics of the datasets used for the constructed suite of benchmarks. **L** denotes the average length of each sample.

movie reviews by the audience, while IMDB contains longer and more professional movie reviews. Therefore, the two datasets can regard each other as OOD data representing a non-semantic shift. For toxic detection, we choose Twitter (Founta et al., 2018) and the Jigsaw dataset from a Kaggle challenge.[4] The Twitter dataset consists of short comments on Tweet, while the Jigsaw dataset consists of longer Wikipedia comments, so they can regard each other as OOD data of the NSS type.

For the SS setting, we consider two tasks where newly emerging classes are common: news topic categorization and dialogue intent classification. For news topic categorization, we choose the AGNews (Corso et al., 2005) and News Category datasets (Misra and Grover, 2021; Misra, 2022) to construct ID/OOD pairs. Specifically, we choose four classes from AGNews and five classes from News Category as ID data and regard the remaining classes from the original datasets as OOD data. For dialogue intent classification, we use the ROSTD (Gangal et al., 2020) and CLINC (Larson et al., 2019) datasets as ID data and regard the annotated unknown intents from the original datasets as OOD data.

A.2 Cross-Task Benchmarks

For the cross-task setting, we follow Zhou et al. (2021) to use SST-2 and 20 Newsgroups (20 NG) (Lang, 1995) as ID data. 20 NG is a news categorization dataset containing 10,182 training samples, 1,132 validation samples, and 7,532 test samples. The average sample length in 20 NG is 289. Naturally, SST-2 and 20 NG can regard each other as OOD data. Besides, we use five additional datasets from different datasets as OOD test data for each ID dataset: TREC-10 (Li and Roth, 2002), WMT-16 (Bojar
RTE (Dagan et al., 2005), and SNLI (Bowman et al., 2015). TREC-10 is a question classification dataset; Multi30k (Elliott et al., 2016) and WMT16 (Bojar et al., 2016) are parts of the English side data of English-German machine translation datasets; RTE (Dagan et al., 2005) and SNLI (Bowman et al., 2015) are the concatenations of the precise and respective hypotheses from NLI datasets. The statistics of the OOD datasets are listed in Table 11.

[4] Available at this link.

Dataset	# Test	L
TREC-10	500	10
Multi30k	1,014	13
WMT16	2,000	22
RTE	3,000	48
SNLI	2,000	21

Table 11: Statistics of OOD datasets in the cross-task setting. **L** denotes the average length of each sample.

Dataset / Loss	$\mathcal{L}_{\text{ce}}$	$\mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{scl}}$	$\mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{margin}}$
SST-2	93.96	94.23	93.69
IMDB	94.56	94.53	94.21
Twitter	93.67	93.64	93.81
Jigsaw	81.82	82.08	82.43
NC	95.39	95.21	95.43
AGNews	91.28	91.03	91.18
ROSTD	99.23	99.21	99.26
CLINC	96.21	96.16	96.08
20 NG	84.52	84.65	84.53

Table 12: Accuracies / F1 scores on the test set of in-distribution data (averaged over five random seeds). We report F1 scores for Twitter and Jigsaw toxic detection and accuracies for other tasks.

A.3 Probing Benchmarks

To probe the linguistic information contained in pre-trained and fine-tuned features, we use two probing tasks designed by Conneau et al. (2018). Each probing dataset contains 100k training samples, 10k validation samples, and 10k test samples. We use the SentEval toolkit (Conneau and Kiela, 2018) along with the recommended hyperparameter space to search for the best probing classifier according to the validation accuracy and report test accuracies.

B Performance on In-Distribution Data

We fine-tune the $\text{RoBERTa}_{\text{base}}$ model on the ID training data to build text classifiers in our main experiments. The model is optimized with the

Adam (Kingma and Ba, 2015) optimizer using a learning rate of 2e-5. We use a batch size of 16 and fine-tune the model for 5 epochs. We evaluate the model on the ID validation set after every epoch and choose the best checkpoint as the final model. The setting is the same for other pre-trained Transformers studied in the paper ($\text{BERT}_{\text{base-uncased}}$ and $\text{RoBERTa}_{\text{large}}$). The performance of fine-tuned $\text{RoBERTa}_{\text{base}}$ models is given in Table 12, where $\mathcal{L}_{\text{ce}}$ denotes the vanilla cross-entropy loss, $\mathcal{L}_{\text{scl}}$ denotes the supervised contrastive loss (Khosla et al., 2020), and $\mathcal{L}_{\text{margin}}$ denotes the margin-based contrastive loss (Zhou et al., 2021). We report the F1 scores on the test set for toxic detection on Twitter and Jigsaw and test accuracies for other tasks.

C Details of OOD Detection Baselines

C.1 Confidence-Based Baselines

Notations In a classification problem with C classes, assume the input is $\mathbf{x}$, we denote $f_i(\mathbf{x})$ is the output logit of class i, and the predicted softmax probability of class i is defined as:

$$p_i(\mathbf{x}) = \max_i \frac{\exp\left(f_i(\boldsymbol{x})\right)}{\sum_{j=1}^{C} \exp\left(f_j(\boldsymbol{x})\right)}. \tag{5}$$

Confidence-based methods obtain the OOD score based the output logits and softmax probabilities.

MSP Hendrycks and Gimpel (2017) propose the maximum softmax probabilty (MSP) baseline, in which the confidence score is defined the predicted maximum softmax probability among C classes:

$$S(\mathbf{x}) = \max_i p_i(\mathbf{x}). \tag{6}$$

Scaling In the ODIN paper (Liang et al., 2018), tempature scaling is applied to the scoring function:

$$S(\mathbf{x}) = \max_i \frac{\exp\left(f_i(\boldsymbol{x})/T\right)}{\sum_{j=1}^{C} \exp\left(f_j(\boldsymbol{x})/T\right)}, \tag{7}$$

where T is the temperature term. Following Hsu et al. (2020), we fix $T = 1000$ in our experiments. Note that ODIN (Liang et al., 2018) also propose an input pre-processing step adding adversarial perturbation to the input image, while we do not use it because it is not directly applicable for discrete inputs in NLP.

Energy Score (Liu et al., 2020) propose to use the free energy function for OOD detection, which is formulated as follows:

$$\begin{aligned} E(\mathbf{x}) &= \sum_{i=1}^{C} e^{f_i(\mathbf{x})}, \\ S(\mathbf{x}) &= -E(\mathbf{x}). \end{aligned} \tag{8}$$

D2U Yilmaz and Toraman (2022) propose to improve out-of-scope detection by exploiting the shape of the entire output distribution. Specifically, the distance of the output distribition $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_c(\mathbf{x}))$ to the uniform distribution U as the OOD score:

$$S(\mathbf{x}) = \text{dst}\left(P\left(x\right), U\right), \tag{9}$$

where dst is the distance function. We use the KL divergence as the distance function as recommended in Yilmaz and Toraman (2022) in our experiments. Note that Yilmaz and Toraman (2022) also propose to use D2U for loss calculation when out-of-scope training data is available, while we do not use it in the training because we follow the mainstream setting in OOD detection works where OOD data is not available for training.

C.2 Density-Based Baselines

PPL Arora et al. (2021) propose to use the token perplexity (PPL) score derived from the GPT-2 language model (Radford et al., 2019) as the OOD score. Following the implementation of Arora et al. (2021),[5] we fine-tune the $\text{GPT-2}_{\text{small}}$ model (117M parameters, similar to $\text{RoBERRa}_{\text{base}}$ in size) for language modeling on the ID training data and use the inverse of the PPL score as the OOD score. Formally, for an input text sequence $\mathbf{x} = \{x_1, \ldots, x_t\}$,

$$\begin{aligned} \text{PPL}(\mathbf{x}) &= \exp\left\{-\frac{1}{t}\sum_{i}^{t} \log p_\theta\left(x_i \mid x_{<i}\right)\right\}, \\ S(\mathbf{x}) &= 1/\text{PPL}(\mathbf{x}), \end{aligned} \tag{10}$$

where t is the number of tokens in $\mathbf{x}$.

C.3 Distance-Based Baselines

Local Outlier Factor (LOF) Lin and Xu (2019) propose to identify unknown user intents by feeding feature vectors derived from LSTM models to the density-based novelty detection algorithm, local outlier factor (LOF) (Breunig et al., 2000). In our implementation, we use the last-layer CLS vector embeddings by the fine-tuned RoBERTa models as the input and train a LOF model following the

[5] Available at this Github repository.

implementation details of Lin and Xu (2019) on the ID training set. Finally, we use the local density output as $S(\mathbf{x})$.

Mahalanobis Distance Detector The Mahalanobis distance detector (MD) (Lee et al., 2018) is a classical distance-based OOD detection method that exploits the sample distance to the nearest ID class in the embedding space to obtain the OOD score. Formally, for a given feature extractor ψ, the Mahalanobis distance score is defined as:

$$S(\mathbf{x}) = -\min_{c\in\Upsilon} (\psi(\mathbf{x}) - \mu_c)^T \Sigma^{-1} (\psi(\mathbf{x}) - \mu_c), \quad (11)$$

where $\Upsilon = \{1, 2, \ldots, C\}$ is the label space containing C classes in the ID task, $\psi(\mathbf{x})$ is the embedding vector of the input $\mathbf{x}$, μ_c is the class centroid for a class c, and Σ is the covariance matrix. The estimations of μ_c and Σ are defined as:

$$\begin{aligned} \mu_c &= \frac{1}{N_c} \sum_{\mathbf{x}\in\mathcal{D}^c_{in}} \psi(\mathbf{x}), \\ \Sigma &= \frac{1}{N} \sum_{c\in\Upsilon} \sum_{\mathbf{x}\in\mathcal{D}^c_{in}} (\psi(\mathbf{x}) - \mu_c)(\psi(\mathbf{x}) - \mu_c)^T, \end{aligned} \quad (12)$$

where $\mathcal{D}^c_{\text{in}} = \{\mathbf{x} \mid (\mathbf{x}, y) \in \mathcal{D}_{\text{in}}, y = c\}$ denotes the training samples belonging to the class c, N is the size of the training set, and N_c is the number of training instances belonging to the class c. As for textual OOD detection based on pre-trained language models, when the feature extractor ψ is the off-the-shelf pre-trained model, i.e. detecting anomalies in the pre-trained feature space (Xu et al., 2021), it is called MD_{pre} in our paper; when ψ is the fine-tuned model, i.e. detecting anomalies in the fine-tuned feature space (Podolskiy et al., 2021), it is called MD_{ft} in our paper.

Contrastive Fine-Tuning Targets Coupled with the MD Detector Zhou et al. (2021) propose to use two forms of contrastive losses to boost textual OOD detection, i.e., the supervised contrastive loss ($\mathcal{L}_{\text{scl}}$) and the margin-based contrastive loss ($\mathcal{L}_{\text{margin}}$). For a classification task containing C classes, given a batch of training examples $\{x_i, y_i\}_{i=1}^M$, where x_i is the input and y_i is the label, the supervised contrastive loss term $\mathcal{L}_{\text{scl}}$ and the final optimization target $\mathcal{L}$ can be formulated as:

$$\begin{aligned} \mathcal{L}_{\text{scl}} &= \sum_{i=1}^{M} \frac{-1}{M|P(i)|} \sum_{p\in P(i)} \log \frac{e^{\boldsymbol{z}_i^\top \boldsymbol{z}_p/\tau}}{\sum_{a\in A(i)} e^{\boldsymbol{z}_i^\top \boldsymbol{z}_a/\tau}}, \\ \mathcal{L} &= \mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{scl}}, \end{aligned} \quad (13)$$

where $A(i) = \{1, ..., M\} \setminus \{i\}$ is the set of all anchor samples, $P(i) = \{p \in A(i) : y_i = y_p\}$ is the set of anchor samples from the same class as i, τ is a temperature hyper-parameter, $\boldsymbol{z}$ is the L2-normalized CLS embedding before the softmax layer, $\mathcal{L}_{\text{ce}}$ is the cross-entropy loss, and λ is a positive coefficient. Following the implementation of Zhou et al. (2021),[6] we use $\tau = 0.3$ and $\lambda = 2$.

The margin-based loss term $\mathcal{L}_{\text{margin}}$ and the final optimization target $\mathcal{L}$ is formulated as:

$$\begin{aligned} \mathcal{L}_{\text{pos}} &= \sum_{i=1}^{M} \frac{1}{|P(i)|} \sum_{p\in P(i)} \|\boldsymbol{h}_i - \boldsymbol{h}_p\|^2, \\ \mathcal{L}_{\text{neg}} &= \sum_{i=1}^{M} \frac{1}{|N(i)|} \sum_{n\in N(i)} \left(\xi - \|\boldsymbol{h}_i - \boldsymbol{h}_n\|^2\right)_+, \\ \mathcal{L}_{\text{margin}} &= \frac{1}{dM} \left(\mathcal{L}_{\text{pos}} + \mathcal{L}_{\text{neg}}\right), \\ \xi &= \max_{i=1}^{M} \max_{p\in P(i)} \|\boldsymbol{h}_i - \boldsymbol{h}_p\|^2, \\ \mathcal{L} &= \mathcal{L}_{\text{ce}} + \lambda \mathcal{L}_{\text{margin}}, \end{aligned} \quad (14)$$

where $N(i) = \{n \in A(i) : y_i \neq y_n\}$ is the set of anchor samples from other classes than y_i, $\boldsymbol{h} \in \mathbb{R}^d$ is the unnormalized CLS embedding before the classification head, ξ is the margin, d is the number of dimensions of $\boldsymbol{h}$, and λ is a positive coefficient. We use $\lambda = 2$ following Zhou et al. (2021).

Except for the optimization target, we use the same hyper-parameters for the two tuning methods as vanilla tuning.

Nearst-Neighbor-Based Detector Sun et al. (2022) explore the efficacy of non-parametric nearest-neighbor distance for OOD detection and show its advantages over the Mahalanobis distance detector on visual OOD detection benchmarks. Specifically, it takes the minus of the average distance from the test sample to the k-nearest training samples in the normalized feature space. We reproduce two variants, i.e., KNN_{pre} using the pre-trained features and KNN_{ft} using the fine-tuned features. We set the neighborhood size $k = 10$ in our experiments.

D Software and Hardware Requirements

We implement our code based on the PyTorch (Paszke et al., 2019) and HuggingFace Transformers (Wolf et al., 2020) Python libraries. All experiments (training and inference) in this paper can be conducted on a single NVIDIA TITAN RTX GPU (24 GB memory), except that the fine-tuning of the $\text{RoBERTa}_{\text{large}}$ model needs 4 TITAN RTX GPUs.

[6] Available at this Github repository

Methods	Avg.	Non-Semantic Shift (NSS)				Semantic Shift (SS)			
		SST-2	IMDB	Twitter	Jigsaw	NC	AGNews	ROSTD	CLINC
MSP (Hendrycks and Gimpel, 2017)	71.62	67.92	74.09	48.75	71.13	75.12	64.84	75.42	95.72
Scaling (Liang et al., 2018)	71.96	67.92	74.09	48.76	71.03	74.60	67.35	75.71	96.20
Energy (Liu et al., 2020)	71.63	69.73	72.84	47.56	69.49	74.19	67.55	76.52	96.18
D2U (Yilmaz and Toraman, 2022)	71.99	67.92	74.09	48.75	71.13	74.62	67.46	75.72	96.26
PPL (Arora et al., 2021)	67.65	79.65	98.51	34.61	**84.36**	57.91	50.67	85.05	50.47
LOF (Lin and Xu, 2019)	76.42	53.39	94.87	62.14	57.88	78.39	70.07	97.49	97.17
MD_{ft} (Podolskiy et al., 2021)	80.39	69.86	90.87	65.83	62.42	3.41	73.51	99.66	**97.57**
MD_{ft} + $\mathcal{L}_{scl}$ (Zhou et al., 2021)	82.22	83.12	88.28	71.13	64.32	81.68	72.74	99.09	97.39
MD_{ft} + $\mathcal{L}_{margin}$ (Zhou et al., 2021)	86.50	93.84	**98.99**	83.04	65.94	82.91	72.00	97.68	97.56
MD_{pre} (Xu et al., 2021)	83.76	**99.99**	98.75	90.59	83.84	57.45	61.53	95.22	82.69
KNN_{ft} (Sun et al., 2022)	81.02	72.00	86.07	74.90	57.33	**84.82**	**75.85**	**99.67**	97.53
KNN_{pre} (Sun et al., 2022)	85.69	**99.99**	98.31	**92.80**	79.53	65.90	69.28	96.89	82.81
GNOME (Ours)	**89.34**	99.98	98.25	89.64	81.10	75.50	73.77	99.63	96.84

Table 13: OOD detection performance (AUROC↑, higher is better) on the developed suites of benchmarks. All values are percentages averaged over five different random seeds, and the best results are highlighted in **bold**. The last column gives the average performance on eight datasets.

E Additional Experimental Results

We display the AUROC results of GNOME and the baselines on the constructed suite of benchmarks in Table 13. The overall trend is consistent with that of the FAR95 results reported in Table 4 in the main text.