Multi-Level Knowledge Distillation for Out-of-Distribution Detection in Text

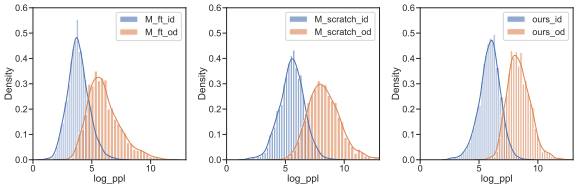
Qianhui Wu[1], **Huiqiang Jiang**[1], **Haonan Yin**[2], **Börje F. Karlsson**[1], **Chin-Yew Lin**[1]
[1] Microsoft [2] Tsinghua University
{qianhuiwu, hjiang, borjekar, cyl}@microsoft.com
yhn21@mails.tsinghua.edu.cn

Abstract

Self-supervised representation learning has proved to be a valuable component for out-of-distribution (OoD) detection with only the texts of in-distribution (ID) examples. These approaches either train a language model from scratch or fine-tune a pre-trained language model using ID examples, and then take the perplexity output by the language model as OoD scores. In this paper, we analyze the complementary characteristics of both OoD detection methods and propose a multi-level knowledge distillation approach that integrates their strengths while mitigating their limitations. Specifically, we use a fine-tuned model as the teacher to teach a randomly initialized student model on the ID examples. Besides the prediction layer distillation, we present a similarity-based intermediate layer distillation method to thoroughly explore the representation space of the teacher model. In this way, the learned student can better represent the ID data manifold while gaining a stronger ability to map OoD examples outside the ID data manifold with the regularization inherited from pre-training. Besides, the student model sees only ID examples during parameter learning, further promoting more distinguishable features for OoD detection. We conduct extensive experiments over multiple benchmark datasets, *i.e.*, CLINC150, SST, ROSTD, 20 NewsGroups, and AG News; showing that the proposed method yields new state-of-the-art performance[1]. We also explore its application as an AIGC detector to distinguish between answers generated by ChatGPT and human experts. It is observed that our model exceeds human evaluators in the *pair-expert* task on the Human ChatGPT Comparison Corpus.

1 Introduction

Machine learning systems such as dialog agents are widely used in many real-world applications.



(a) M_{finetune} (b) $M_{\text{fromScrach}}$ (c) Ours

Figure 1: Visualization of OoD-score distribution of both ID and OoD examples[2]. Less overlap is preferred.

These systems have proved to work well when the distributions of training data and test data are the same or closely similar. However, when there is a gap between training distribution and test distribution, trained models may generate dubious, and even disastrous, predictions that could cause serious AI safety issues (Hendrycks and Gimpel, 2017). Therefore, it is crucial to detect out-of-distribution (OoD) inputs for deployed machine learning systems. Moreover, lifelong learning systems are usually required to discover OoD examples during their application to create new tasks and learn them incrementally (Liu and Mazumder, 2021), which further highlights the importance of OoD detection. In this paper, we focus on the task of OoD detection with only in-distribution texts available during learning for its capability of dealing with diverse scenarios such as non-classification applications while requiring the least data collection effort.

Recent studies have well demonstrated the validity of self-supervised representation learning (Manolache et al., 2021; Arora et al., 2021; Mai et al., 2022). These approaches use ID examples to either fine-tune a large pre-trained language model (M_{finetune}) (Manolache et al., 2021; Mai et al., 2022) or to train a language model from scratch (Mai et al., 2022) ($M_{\text{fromScratch}}$). Given an input sequence for inference, token perplexity output by the learned/fine-tuned language model is regarded

[1]Our code is available at https://github.com/microsoft/KC/tree/main/papers/MLKD_OOD.

[2]We use test data of Group 2 (SST). Please refer to 4.1 for more details.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 7317–7332
July 9-14, 2023

as the OoD score, *i.e.*, indication of an example being OoD. However, both methods have limitations. For $\mathrm{M}_{\mathrm{finetune}}$, since the pre-training corpus usually consists of huge-scale datasets from a diverse range of genres, it is possible that some OoD examples are seen during pre-training, leading to a risk of non-distinguishing perplexities between ID examples and these "leaked" OoD examples as shown in Figure 1a. This issue is eliminated in $\mathrm{M}_{\mathrm{fromScratch}}$ which only sees ID examples during training. However, using only ID examples to minimize the self-supervised language modeling loss without any other constraints may result in a less compact representation space of ID data. Consequently, OoD examples have more chance to locate in the ID data manifold, leading to the overlap between perplexity distributions of ID examples and OoD examples as shown in Figure 1b.

Inspired by Ma et al. (2022), which indicates that unsupervisedly trained sentence embeddings (mean pooling over all token representations) (Giorgi et al., 2021) can achieve non-trivial performance in the sentence classification task, here we contemplate that the pre-training procedure of language models can facilitate their ability of capturing semantic relatedness. In other words, language models are promoted to map examples with different semantics to different manifolds via pre-training. Therefore, we suggest inheriting the representation space with such characteristics gained from pre-training to mitigate the limitation of $\mathrm{M}_{\mathrm{fromScratch}}$.

In this paper, we propose to adopt multi-level knowledge distillation to integrate the strengths from both methods while mitigating their limitations. Specifically, we first produce a teacher model by fine-tuning a large pre-trained language model with ID training examples, so that features of the teacher model can well represent the ID data manifold, while to some extent preserving the ability to map examples with different semantics to different manifolds. Then, we perform knowledge distillation to learn a student model from scratch, using ID training examples with supervision from the fine-tuned teacher model. To learn the teacher's representation space more thoroughly, we not only perform prediction layer distillation, but also propose a similarity-based intermediate layer distillation method to make the student model aware of the information flow inside the teacher's layers. Finally, we deploy the learned student model to compute token perplexity for each inference example as its OoD score and compare it with a threshold to determine whether it is OoD or not. In contrast to $\mathrm{M}_{\mathrm{finetune}}$, our student model doesn't see any OoD examples during parameter learning, thus avoiding the leakage of OoD examples. Compared with $\mathrm{M}_{\mathrm{fromScratch}}$, our student model is trained with the regularization inherited from pre-training via the multi-level supervision from the teacher model, thus gaining a stronger ability to map OoD examples outside the ID data manifold. Both are conducive to more distinguishable representations for OoD detection.

Moreover, with the development of automatic text generation technologies such as InstructGPT (Ouyang et al., 2022) and ChatGPT[3], the risk of automatically generated content to society (*e.g.*, generating fake news or fake reviews of products) is increasing. Therefore, we further adapt our model to distinguish texts generated by AI models and human experts. By conducting experiments on the Human ChatGPT Comparison Corpus (HC3), we observe that our model beats human evaluators and shows excellent capability in the *pair-expert* task.

Our major contributions can be summarized as:

- We analyze the limitations of existing methods for OoD detection with solely ID examples. We investigate their complementary characteristics and propose a novel multi-level knowledge distillation-based approach to unify the strengths of previous studies while mitigating their limitations. To our best knowledge, this is the first attempt to adapt knowledge distillation to textual OoD detection.

- We propose a dynamic intermediate layer distillation method to force the student model to thoroughly explore the representation space of the teacher model. The learned student can well represent the ID data manifold while gaining a stronger ability to map OoD examples outside the ID data manifold.

- Prior studies have conducted experiments on different benchmarks and do not directly compare to each other (Xu et al., 2021; Arora et al., 2021; Manolache et al., 2021; Mai et al., 2022; Gangal et al., 2020). We compare our approach to previous state-of-the-art methods on multiple datasets across genres and domains, *i.e.*, CLINC150 (Larson et al., 2019),

[3] https://openai.com/blog/chatgpt/

SST (Socher et al., 2013), ROSTD (Gangal et al., 2020), 20 NewsGroups (Lang, 1995), and AG News (Zhang et al., 2015); showing that the proposed method yields new state-of-the-art performance.

- We apply our model as an AIGC detector to distinguish automatically generated texts from those generated by human experts. The experimental results show that our model outperforms human evaluators in the *pair-expert* task on the HC3 benchmark.

2 Related Work

Considering the accessibility of OoD data and class labels of ID data, previous work for OoD detection can be divided into three categories: i) OoD data available; ii) OoD data unavailable, but class labels of ID examples available; and iii) both types of data unavailable.

Methods *with* supervision from OoD data. These methods usually train a binary classifier (Larson et al., 2019) or a multi-class classifier (Hendrycks et al., 2019; Zhan et al., 2021) to detect OoD examples, where OoD data is regarded as an independent class for training. OoD data used as supervision is collected from other existing datasets that are disjoint with the ID training data (Hendrycks and Gimpel, 2017). Some previous work also introduces synthesized pseudo outliers to try to find a more representative classification hyperplane for OoD data (Zhan et al., 2021). Since there are various reasons for an example to be considered OoD, *e.g.*, being out-of-domain (Daumé III, 2007), infrequent (Sagawa et al., 2020), or adversarial (Carlini and Wagner, 2017; Arora et al., 2021), it is impractical to collect OoD data for learning.

Methods *without* supervision from OoD data but *with* supervision from ID class labels. These approaches generally consider the scenario of multi-class classification such as intent detection (Lin and Xu, 2019; Yilmaz and Toraman, 2020) and assume that class labels of in-distribution (ID) data are available during model training. Class probabilities (Hendrycks and Gimpel, 2017; Shu et al., 2017; Liang et al., 2018; Zeng et al., 2021b; Zhou et al., 2021) and distance or density in latent space (Lin and Xu, 2019; Xu et al., 2020; Podolskiy et al., 2021; Zeng et al., 2021a; Zhou et al., 2021) are the most prevalent metrics. In particular, Hendrycks and Gimpel (2017) propose a strong baseline which takes the maximum softmax probability (MSP) of a multi-class classifier as a measure of OoD score. Based on that, lots of following studies devote to optimizing the model's calibration with temperature scaling (Liang et al., 2018), contrastive learning (Zeng et al., 2021b; Zhou et al., 2021), *etc*. For distance and density based approaches, they first learn discriminative deep features via carefully designed loss functions, *e.g.*, large margin cosine loss (Lin and Xu, 2019; Xu et al., 2020) and contrastive loss (Zeng et al., 2021a,b; Zhou et al., 2021). Then, compute distance or density metrics such as local outlier factor (LOF) (Breunig et al., 2000; Lin and Xu, 2019; Zeng et al., 2021b) and Gaussian discriminant analysis (Xu et al., 2020; Podolskiy et al., 2021; Zeng et al., 2021a,b) to detect OoD examples.

Methods *without* supervision from both OoD data nor ID class labels. Given the in-distribution data, these methods generally estimate ID density and regard test examples that deviate from the estimated distribution as OoD examples. Previous work for this setting mainly focuses in the field of computer vision. Variational autoencoders (VAE) (Kingma and Welling, 2014) and generative adversarial networks (GAN) are frequently taken as the backbone models for density estimation (Chen et al., 2018; Zenati et al., 2018). In natural language processing, current studies generally perform self-supervised language modeling on ID examples and take token perplexity as OoD score (Arora et al., 2021; Manolache et al., 2021; Mai et al., 2022). Gangal et al. (2020) further introduces an independent background model to correct confounding background statistics. Moreover, Xu et al. (2021) learn a combination of latent representations from different layers of pre-trained transformers to represent ID data manifold in a compact way. Based on that, one-class classification methods such as one-class SVM (Schölkopf et al., 2001) and SVDD (Tax and Duin, 2004) can be used to detect OoD examples. Jin et al. (2022) combine unsupervised clustering and contrastive learning to learn the ID data distribution and further use Gaussian mixture model (GMM) (Reynolds, 2009) for density estimation.

Our approach falls closer to the last category. We propose an intermediate layer distillation method and adopt multi-level knowledge distillation to unify the strengths of different language modeling likelihood-based methods, while mitigating their

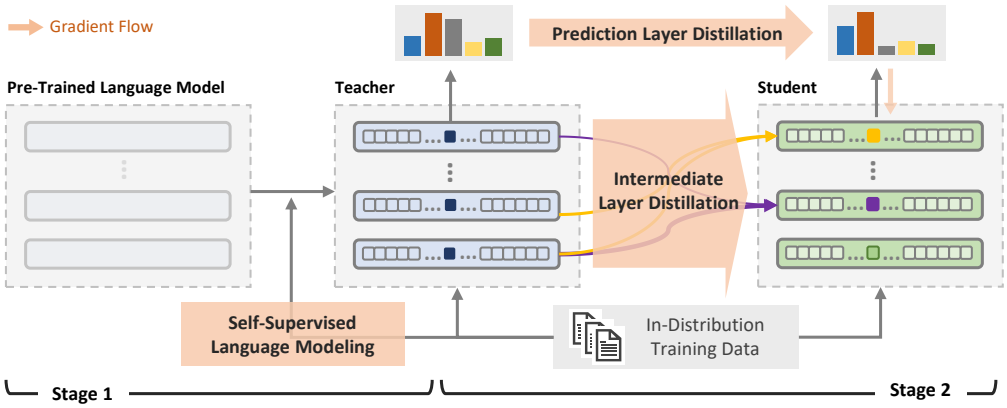


Figure 2: Framework of the proposed approach.

limitations. To our best knowledge, this is the first attempt to adapt knowledge distillation to textual OoD detection. Compared with Xu et al. (2021), our approach does not involve any hyper-parameter sensitive one-class classification stage. Compared with Jin et al. (2022), our proposed method requires no prior knowledge about the ID data, *e.g.*, the number of semantic categories. Moreover, our approach is orthogonal to Gangal et al. (2020) and we can combine both to achieve better performance.

3 Methodology

In this section, we elaborate on the proposed multi-level knowledge distillation approach for OoD detection. First, we clarify how to produce a teacher model that estimates the distribution of ID data. Then, we describe the proposed multi-level knowledge distillation procedure to teach a randomly initialized student model via the produced teacher model. Note that in this paper, both the teacher and student networks are built with Transformer layers (Vaswani et al., 2017). Figure 2 illustrates the overall framework.

3.1 Teacher Model

Here, we use language models as the base model to estimate the distribution of ID data. Causal language modeling (CLM) and masked language modeling (MLM) are the most representative language models. CLM predicts the next token based on unidirectional contexts, while MLM first masks some tokens and then predicts the masked tokens conditioned on bidirectional context. Since the nature of MLM requires the model to forward multiple times so that the probability of each token in the sentence could be predicted, it is time-consuming to exploit MLM to estimate ID data distribution. Therefore, we utilize CLM in our approach.

Given a text sequence $\boldsymbol{x} = \{x_i\}_{i=1}^{N}$, where x_i is the i-th token and N is the sequence length, the probability estimation function of CLM can be formulated as:

$$p(\boldsymbol{x}) = \prod_{i=1}^{N} p(x_i|x_{<i}), \tag{1}$$

where $x_{<i}$ denotes tokens before the i-th token x_i.

In this paper, we fine-tune a large pre-trained language model on the ID training examples to produce the teacher model. The loss function *w.r.t.* $\boldsymbol{x}$ is:

$$\mathcal{L}^{\boldsymbol{x}}(\theta_{tea}) = -\frac{1}{N}\sum_{i=1}^{N} \log p(x_i|x_{<i};\theta_{tea}), \tag{2}$$

where θ_{tea} represents the parameters of the teacher model.

3.2 Knowledge Distillation

With the supervision from this teacher model, we then train a student model by performing both prediction layer distillation and intermediate layer distillation.

3.2.1 Prediction Layer Distillation

Given a training ID sequence $\boldsymbol{x} = \{x_i\}_{i=1}^{N} \in \mathcal{D}_{in}$, the learning loss for prediction layer distillation *w.r.t.* $\boldsymbol{x}$ is formulated as the Kullback-Leibler divergence between the output probability distributions over the vocabulary $\mathcal{V}$ output by the teacher model and by the student model. Averaging over all tokens, we have:

$$\mathcal{L}^{\boldsymbol{x}}_{pred}(\theta_{stu}) = -\frac{1}{N}\sum_{i=1}^{N}\mathrm{KL}\left(p(x_i|x_{<i};\theta_{tea}), p(x_i|x_{<i};\theta_{stu})\right), \quad (3)$$

where x_i represents the i-th token in $\boldsymbol{x}$, $p(x_i|x_{<i};\theta_{tea})$ denotes the probability distribution for the i-th token output by the teacher model, and $p(x_i|x_{<i};\theta_{stu})$ represents that of the student model.

3.2.2 Intermediate Layer Distillation

Considering that different layers in large pre-trained language models generally correspond to features at various abstraction levels (Jawahar et al., 2019; Caucheteux et al., 2021), here we propose an intermediate layer distillation method to facilitate the student model acquiring a more comprehensive awareness of the information flow inside the teacher's layers. Instead of pre-defining a fixed mapping function between teacher layers and student layers, we dynamically match each hidden vector of the student to multiple hidden vectors of different layers of the teacher.

Specifically, we first use ℓ_2 distance to measure the similarity between the hidden vector produced by the student model *w.r.t.* the i-th token at the l-th layer (*i.e.*, $h^{stu}_{l,i}$) and that produced by the teacher model *w.r.t.* the i-th token at the j-th layer (*i.e.*, $h^{tea}_{j,i}$) :

$$s_{l,i}(j) = -\left\|h^{stu}_{l,i} - W_j h^{tea}_{j,i}\right\|_2, \quad (4)$$

where $j \in \mathcal{A}$, $\mathcal{A}$ represents the set of the teacher's layer indexes, and W_j are learnable parameters.

Let $\mathcal{S}^K_{l,i} = \{s^k_{l,i}(\cdot)\}^K_{k=1}$ denote the top-K similarities computed by Eq. (4) *w.r.t.* $h^{stu}_{l,i}$. We then train the student model by maximizing the similarities in $\mathcal{S}^K_{l,i}$. Let β_k denote the to-be-learned weighting scalar corresponding to the k-th similarity in $\mathcal{S}^K_{l,i}$. The learning loss at the l-th layer *w.r.t.* $\boldsymbol{x}$ can be formulated as:

$$\mathcal{L}^{\boldsymbol{x}}_{(l)}(\theta_{stu}) = \frac{1}{N}\frac{1}{K}\sum_{i=1}^{N}\sum_{k=1}^{K} -\beta_k \cdot s^k_{l,i}(\cdot). \quad (5)$$

Finally, we integrate the prediction layer distillation and the intermediate layer distillation. Let $\mathcal{T}$ denote the set of the student's layer indexes, the whole training loss of the student model is the summation of losses *w.r.t.* all sentences in $\mathcal{D}_{in}$:

$$\mathcal{L}(\theta_{stu}) = \sum_{\boldsymbol{x}\in\mathcal{D}_{in}}\left(\lambda\mathcal{L}^{\boldsymbol{x}}_{pred} + (1-\lambda)\sum_{l\in\mathcal{T}}\mathcal{L}^{\boldsymbol{x}}_{(l)}\right), \quad (6)$$

where λ is a hyper-parameter for weighting.

3.2.3 Inference

For inference, we only use the learned student model θ_{stu} to compute perplexity for each token x_i in an input sequence $\boldsymbol{x} = \{x_i\}^N_{i=1}$. We calculate the OoD score *w.r.t.* $\boldsymbol{x}$ by averaging over all tokens:

$$\mathrm{score}(\boldsymbol{x}) = -\frac{1}{N}\sum_{i=1}^{N}\log p(x_i|x_{<i};\theta_{stu}). \quad (7)$$

We define a threshold γ to classify OoD examples against ID examples. Specifically, $\boldsymbol{x}$ is predicted as an OoD example if $\mathrm{score}(\boldsymbol{x}) > \gamma$, else it is an ID example.

4 Experiments

4.1 Settings

Datasets Following Xu et al. (2021), Manolache et al. (2021), and Gangal et al. (2020), we conduct five groups of experiments for evaluation: **CLINC150** (Larson et al., 2019), **SST** (Socher et al., 2013), **ROSTD** (Gangal et al., 2020), **20NewsGroups** (Lang, 1995), and **AG-News** (Zhang et al., 2015). For dataset statistics and other detailed information, please refer to Appendix A.1.

Evaluation Following Hendrycks and Gimpel (2017) and Jin et al. (2022), we utilize the Area Under Operating Characteristic curve (AUROC), the Area Under Precision-Recall curve (AUPR), and the False Alarm Rate (*i.e.*, False Positive Rate) at 95% Recall (FAR95) as metrics for a comprehensive evaluation. Since our target is to detect OoD examples without having to rely on the semantic category of ID data, we treat OoD as the positive class for computing AUROC, AUPR, and FAR95.

Implementation Details We implement our approach based on PyTorch 1.12.0[4] and HuggingFace's Transformers[5]. For fair comparison, we utilize `GPT2-small` (Radford et al., 2019) as the base model as it has a similar number of parameters as BERT-base (Devlin et al., 2019) used in Xu et al. (2021) and the discriminator of ELECTRA (Clark et al., 2020) used in Manolache et al. (2021). Following Xu et al. (2021), we train both the teacher model and the student model for 5 epochs on CLINC150, SST, and ROSTD. Following Manolache et al. (2021), we train our models for 30 epochs on 20 NewsGroups and 5 epochs

[4] https://pytorch.org/

[5] https://github.com/huggingface/transformers

Method	CLINC150			SST			ROSTD		
	AUROC (↑)	AUPR (↑)	FAR95 (↓)	AUROC (↑)	AUPR (↑)	FAR95 (↓)	AUROC (↑)	AUPR (↑)	FAR95 (↓)
TF-IDF+SVD†	58.5	21.8	-	78.0	73.2	-	-	-	-
Likelihood Ratios‡	-	-	-	-	-	-	96.35±0.41	93.44±0.37	20.10±5.25
MDF+IMLM†	77.8	39.1	-	93.6	89.4	-	-	-	-
MDF+IMLM	77.46±0.33	39.23±0.52	65.87±1.13	96.79±0.06	95.62±0.06	11.68±0.41	97.71±0.10	93.00±0.32	9.03±0.43
DATE	83.38±0.15	50.21±0.18	66.67±1.65	82.20±0.18	83.11±0.41	55.26±1.97	96.59±0.43	91.77±1.06	17.06±1.82
$M_{finetune}$	89.76±0.13	62.39±0.29	33.77±0.91	92.67±0.19	91.93±0.17	33.67±1.21	98.67±0.04	97.47±0.09	6.27±0.30
$M_{fromScratch}$	91.73±0.12	68.78±0.62	28.31±0.40	96.60±0.65	96.42±0.63	17.98±3.47	99.10±0.03	98.25±0.06	3.88±0.22
Ours	**92.51±0.18**	**70.94±0.78**	**27.16±0.65**	**97.97±0.40**	**97.81±0.42**	**9.50±2.09**	**99.14±0.03**	**98.33±0.06**	**3.79±0.11**

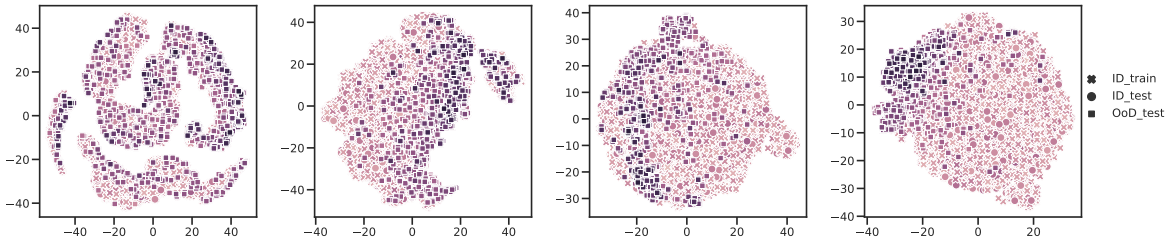
Table 1: Performance comparison on CLINC150, SST, and ROSTD. † and ‡ represents results reported in Xu et al. (2021) and Gangal et al. (2020), respectively.

Method	comp			rec			sci		
	AUROC (↑)	AUPR (↑)	FAR95 (↓)	AUROC (↑)	AUPR (↑)	FAR95 (↓)	AUROC (↑)	AUPR (↑)	FAR95 (↓)
IsoForest†	66.1	-	-	59.4	-	-	57.8	-	-
OCSVM†	78.0	-	-	70.0	-	-	64.2	-	-
CVDD†	74.0	-	-	60.6	-	-	58.2	-	-
DATE†	92.1	-	-	83.4	-	-	69.7	-	-
DATE	92.04±0.14	97.05±0.38	46.56±1.37	83.09±0.22	95.46±0.74	65.62±2.17	66.30±0.16	90.64±0.12	**81.68±0.67**
MDF + IMLM	86.37±0.12	94.08±0.08	53.33±0.36	75.77±0.25	90.63±0.13	69.63±0.25	67.02±0.12	87.81±0.10	86.18±0.68
$M_{finetune}$	87.60±0.22	95.14±0.09	54.98±0.88	74.41±0.25	92.45±0.06	73.04±0.31	63.15±0.24	88.89±0.11	86.21±0.34
$M_{fromScratch}$	91.29±0.60	97.29±0.11	49.36±5.08	85.36±0.56	96.27±0.12	64.28±3.66	67.80±0.23	90.40±0.16	85.12±0.23
Ours	**92.41±0.22**	**97.49±0.12**	**43.30±1.55**	**87.68±0.15**	**96.79±0.08**	**54.66±1.41**	**69.83±0.29**	**91.14±0.15**	84.95±0.54

Method	misc			pol			rel		
	AUROC (↑)	AUPR (↑)	FAR95 (↓)	AUROC (↑)	AUPR (↑)	FAR95 (↓)	AUROC (↑)	AUPR (↑)	FAR95 (↓)
IsoForest†	62.4	-	-	65.3	-	-	71.4	-	-
OCSVM†	62.1	-	-	76.1	-	-	78.9	-	-
CVDD†	75.7	-	-	71.5	-	-	78.1	-	-
DATE†	86.0	-	-	81.9	-	-	86.1	-	-
DATE	82.25±0.12	98.93±0.01	66.81±2.40	81.74±0.16	96.72±0.10	64.52±2.72	86.14±0.09	97.66±0.02	62.86±0.93
MDF + IMLM	62.26±7.70	96.80±0.81	92.81±5.10	81.05±0.16	95.48±0.06	62.38±0.31	80.85±0.26	96.14±0.08	65.14±0.35
$M_{finetune}$	83.27±0.11	98.93±0.02	56.28±0.72	75.37±0.33	95.42±0.09	71.32±0.44	75.75±0.27	95.82±0.07	75.53±0.49
$M_{fromScratch}$	85.01±0.25	99.16±0.01	63.56±2.00	87.53±0.17	97.87±0.01	51.32±0.62	86.48±0.18	97.90±0.06	58.57±1.23
Ours	**89.02±0.24**	**99.35±0.03**	**44.66±0.73**	**88.47±0.15**	**98.11±0.04**	**50.85±1.07**	**87.51±0.11**	**98.01±0.03**	**55.74±0.92**

Table 2: Performance comparison on 20NewsGroups. † represents results reported in Manolache et al. (2021).

on AG News, respectively. We use a batch size of 8 and a maximum sequence length of 128. For the optimizers, we use AdamW (Loshchilov and Hutter, 2019) with the learning rate set to $5e-5$ for all models. λ in Eq. (6) is set to 0.5. Motivated by Haidar et al. (2022), which indicate that using only intermediate layers for distillation (from RoBERTa_24 to RoBERTa_6) works the best, we only distill the intermediate layers ($\mathcal{T} = \{l\}_{l=3}^{9}$). We compare each student layer to a combination of K teacher layers, as Haidar et al. (2022) show that concatenated representation distillation of sorted randomly selected K intermediate layers is superior to layer-wise distillation. We choose $K = 2$ for the cardinality of the similar set $\mathcal{S}_{l,i}^{K}$ considering that there's no information fusion among different teacher layers if $K = 1$ and that a larger K may introduce too much noise due to the weighted average of representations as in Equation (5). We re-implement Xu et al. (2021) and Manolache et al. (2021) with BERT-base using their open-sourced code, and report the results on all benchmark datasets for a more comprehensive comparison. All experiments are conducted on one Tesla V100 (16GB). The trainable parameters (*i.e.*, θ_{tea} and θ_{stu}) are 248M. The training time is about 30 minutes for each model.



(a) Pre-trained GPT-2 (b) $\mathrm{M}_{\mathrm{finetune}}$ (c) $\mathrm{M}_{\mathrm{fromScrath}}$ (d) Ours

Figure 3: T-SNE Visualization of sentence representations from different models. Darker colors represents higher OoD scores.

4.2 Main Results

Tables 1 and 2 report the results of our approach alongside those reported by previous state-of-the-art methods on CLINC150, SST, ROSTD, and 20NewsGroups. It can be seen that our proposed method outperforms the prior methods with a large margin in most experiments, achieving an improvement of up to 9.13, 20.73, 38.71 points in terms of AUROC, AUPR, and FAR95, respectively, on CLINC150. This well demonstrates the effectiveness of the proposed approach.

The results also show that $\mathrm{M}_{\mathrm{fromScrath}}$ generally leads to superior performance than $\mathrm{M}_{\mathrm{finetune}}$. We conjecture that seeing no OoD examples during parameter learning helps the randomly initialized model avoid optimizing toward OoD distribution. Without the bias and constraints inherited from the pre-training process, the model trained from scratch is more likely to find a local minimum that better fits the ID training text and thus leads to more distinguishable features for OoD detection. Moreover, our approach, which uses the fine-tuned model to teach a randomly initialized model, can integrate their strengths via the proposed multi-level knowledge distillation process, resulting in superior performance.

4.3 Ablation Study

To validate the contributions of different components in the proposed approach, here we introduce two variants of our model for ablation study: i) *Ours w/ GPT2_Init_*θ_{stu}, which initializes the student model with the pre-trained GPT-2 model. ii) *Ours w/o* $\mathcal{L}^{\boldsymbol{x}}_{(l)}$, which eliminates the loss *w.r.t.* intermediate layer distillation and only conducts output layer distillation to learn the student model. Table 3 shows the results.

Comparing Ours with *Ours w/ GPT2_Init_*θ_{stu}, we can see that replacing the randomly initialized student model with a pre-trained student model will cause a significant performance drop, well verifying our motivation to incorporate $\mathrm{M}_{\mathrm{fromScratch}}$ with $\mathrm{M}_{\mathrm{finetune}}$. Table 3 also illustrates that removing the constraints on intermediate layers, *i.e.*, $\mathcal{L}^{(\boldsymbol{x})}_{(l)}$, the student model's performance will decrease by 0.90, 0.87, and 5.03 in terms of AUROC, AUPR, and FAR95, respectively. This well validates both the effectiveness and necessity of intermediate layer distillation. Moreover, though eliminating the intermediate distillation, the student model in *Ours* - $\mathcal{L}^{\boldsymbol{x}}_{(l)}$ which is derived with only the prediction layer distillation still outperforms the baseline model $\mathrm{M}_{\mathrm{fromScratch}}$. We owe this superiority to the more informative supervision, *i.e.*, the probability distribution produced by the teacher model, compared with the ground-truth one-hot supervision used in $\mathrm{M}_{\mathrm{fromScratch}}$.

	AUROC (↑)	AUPR (↑)	FAR95 (↓)
Ours	**97.97**±0.40	**97.81**±0.42	**9.50**±2.09
Ours w/ GPT2_Init_θ_{stu}	94.12±0.60	94.21±0.64	31.72±3.20
Ours w/o $\mathcal{L}^{\boldsymbol{x}}_{(l)}$	97.07±0.23	96.94±0.23	14.53±1.05

Table 3: Ablation study on SST.

4.4 Analysis on Distribution of Sentence Repr.

To bring up insights on how multi-level knowledge distillation promotes OoD detection, we utilize *t-SNE* (Van der Maaten and Hinton, 2008) to reduce the dimension of sentence representations obtained from pre-trained GPT-2, $\mathrm{M}_{\mathrm{finetune}}$, $\mathrm{M}_{\mathrm{fromScratch}}$, and the student model in our approach. Here, we produce sentence representations by averaging token representations. The visualization is shown in Figure 3. In Figure (3a), we can see that ID and OoD examples locate uniformly in several *separate* data manifolds because the model has no sense of what is OoD and thus struggles to distinguish OoD

examples from ID examples. After applying fine-tuning to the pre-trained model, representations of the ID data converge to fewer manifolds and it becomes easier to classify ID and OoD examples, specifically, OoD examples mainly lie in the right-side of Figure (3b), which might lead to easier separation in high dimension space. However, when comparing Figures (3b) and (3c), we can notice that M_{finetune} doesn't fit the ID examples as well as $M_{\text{fromScratch}}$ because the representation distribution of ID examples in Figure (3c) is more condensed. This supports our conjecture that the pre-training process may guide the model to a position near an inferior local minimum where ID and OoD examples are less separable. Last but not least, Figure (3d) indicates that our student model produces a more compact distribution for ID examples when trained from scratch. Meanwhile, Figure (3d) also shows that ID representations and OoD representations produced by our student model are more dissociable. We conjecture that this is because the model gains some ability to separate examples of different semantics via knowledge distillation - the teacher model equips this knowledge during pre-training.

4.5 Application

ChatGPT, an optimized language model for dialog[6], has attracted great attention in the NLP field since its inception. It is capable of providing fluent and comprehensive responses to a large variety of questions. To study how close ChatGPT is to human experts, Guo et al. (2023) proposed the Human ChatGPT Comparison Corpus (HC3), where each question is paired with two answers, one is a human answer collected from wiki sources and public QA datasets, and the other is generated by ChatGPT[7]. By conducting human evaluation, Guo et al. (2023) indicates that it can be difficult to distinguish texts generated by ChatGPT from those provided by human experts, and further propose a RoBERTa (Liu et al., 2019) based detector to distinguish both.

Following Guo et al. (2023), in this section, we adapt our model as an AI-generated content (AIGC) detector to explore its capability for preventing the potential risks of AIGC abuse. As our model uses perplexity as the OoD score and Guo et al. (2023) reveal that ChatGPT-generated answers are usually of low perplexities, here we take ChatGPT-generated answers as in-distribution data to train our model. We divide the in-distribution data into a training set and a test set. We use all the human-generated answers as the OoD test set.

We first evaluate our model as in 4.1 and Table 4 shows its performance results. We can see that our approach significantly outperforms prior state-of-the-art methods *DATE* and *MDF+IMLM* under the same settings. Surprisingly, our unsupervised method demonstrates comparable performance with *RoBERTa-single Detector*, which is a RoBERTa-based sentence classifier trained with the supervision from all the ChatGPT-generated and human-generated texts.

	AUROC (↑)	AUPR (↑)	FAR95 (↓)
Unsupervised methods:			
DATE	75.80	91.20	85.15
MDF+IMLM (BERT)	89.61	96.80	42.35
MDF+IMLM (GPT2-small)	91.53	92.56	31.84
Ours	**99.80**	**99.95**	**0.61**
Supervised method:			
chatgpt-detector-roberta[8]	99.98	99.99	0.04

Table 4: Performance comparison on HC3.

We also compare our model to the human evaluation results listed in Guo et al. (2023). Given two answers corresponding to the same question, with one being generated by ChatGPT and the other by a human expert, our model is required to determine which answer is generated by ChatGPT. Table 5 shows that our model beats human evaluators and perfectly handles this task.

	Human	Ours
All	0.90	**1.00**
reddit_eli5	0.97	-
open_qa	0.98	-
wiki_csai	0.97	-
medical	0.97	-
finance	0.79	-

Table 5: Accuracy comparison with human evaluation on paired answers to determine the ChatGPT-generated responses. Note that we run experiments using the whole test set, while human evaluation in Guo et al. (2023) is conducted on a subset of it.

[6] https://chat.openai.com/

[7] HC3 covers a wide range of domains (open-domain, computer science, finance, medicine, law, and psychology) and is widely used in lots of recent studies.

[8] https://github.com/Hello-SimpleAI/chatgpt-comparison-detection

Method	business			sci		
	AUROC (↑)	AUPR (↑)	FAR95 (↓)	AUROC (↑)	AUPR (↑)	FAR95 (↓)
IsoForest†	73.2	-	-	76.9	-	-
OCSVM†	83.2	-	-	80.7	-	-
CVDD†	79.6	-	-	79.0	-	-
DATE†	90.1	-	-	84.0	-	-
DATE	89.46±0.15	95.19±0.12	50.26±1.49	83.88±0.37	93.29±0.24	60.97±3.12
MDF + IMLM	90.12±0.06	95.36±0.03	34.81±0.39	**85.93±0.22**	**94.62±0.09**	54.37±1.30
M_{finetune}	89.19±0.10	95.27±0.05	67.63±0.24	76.52±0.14	88.36±0.13	63.95±0.42
$M_{\text{fromScratch}}$	91.49±0.16	96.29±0.08	32.56±0.87	83.76±0.09	92.12±0.07	54.42±0.31
Ours	**92.38±0.11**	**96.72±0.05**	**29.94±0.68**	85.12±0.19	92.92±0.16	**51.18±0.38**
Method	sports			world		
	AUROC (↑)	AUPR (↑)	FAR95 (↓)	AUROC (↑)	AUPR (↑)	FAR95 (↓)
IsoForest†	84.7	-	-	79.6	-	-
OCSVM†	92.4	-	-	79.9	-	-
CVDD†	89.9	-	-	84.0	-	-
DATE†	95.9	-		90.0	-	-
DATE	96.01±0.22	98.48±0.08	19.21±0.68	90.08±0.14	96.04±0.06	42.46±0.38
MDF + IMLM	**97.91±0.06**	**99.20±0.03**	**7.37±0.23**	**91.28±0.08**	**96.68±0.04**	38.63±0.47
M_{finetune}	91.46±0.19	96.09±0.11	30.79±0.52	84.19±0.07	92.17±0.05	48.24±0.42
$M_{\text{fromScratch}}$	97.00±0.08	98.86±0.04	10.75±0.39	89.46±0.05	95.27±0.04	38.51±0.59
Ours	97.31±0.06	98.97±0.03	9.27±0.36	89.89±0.04	95.43±0.04	**37.86±0.35**

Table 6: Performance comparison on AGNews. † represents results reported in Manolache et al. (2021).

5 Conclusion

In this paper, we focus on the setting of OoD detection without supervision from both OoD data nor ID class labels. We analyze the complementary characteristics of existing self-supervised representation learning-based methods and propose a multi-level knowledge distillation approach to integrate their strengths, while mitigating their limitations. We evaluate the proposed method on multiple datasets and results show that the proposed method yields new state-of-the-art performance. We analyze why our approach attains superior performance by conducting ablation studies and sentence representation visualization. We further apply our model as an AIGC detector to distinguish ChatGPT-generated texts from those generated by human experts and the experimental results demonstrate that our model outperforms human evaluators in the setting of paired answers.

Limitations

Table 6 shows the results of our model and other methods on the AGNews benchmark. Interestingly, we notice that our approach reports a slightly inferior performance when compared with *MDF+IMLM* (Xu et al., 2021). We can see that methods using sentence representations based on token aggregation, *e.g.*, fastText[9] or Glove (Pennington et al., 2014)-based IsoForest, OCSVM, and CVDD (Ruff et al., 2019), as well as BERT based MDF + IMLM (Xu et al., 2021), perform especially well on AGNews compared to their performance on other datasets. We conjecture that this is because AGNews has a much larger variation of sequence length (36.6) than other datasets (around 7 or 8). A larger length variation will lead to more acute fluctuations in perplexities, especially when adopting an autoregressive language model with unidirectional context such as GPT-2-small in this paper, making it more difficult to distinguish between ID and OOD examples than in other datasets. In contrast, sentence representation based methods benefit from directly estimating the OoD score using the information from the whole sentence, thus producing superior performance. Fortunately, the limitation of auto-regressive modeling could be eliminated by leveraging Transcormer (Song et al., 2022) as the base model of our approach, where bidirectional context is used for estimating tokens at each position. We leave this for future work.

[9] https://github.com/facebookresearch/fastText

References

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*, pages 10687–10701, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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 *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 93–104.

Nicholas Carlini and David Wagner. 2017. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 3–14.

Charlotte Caucheteux, Alexandre Gramfort, and Jean-Rémi King. 2021. Gpt-2's activations predict the degree of semantic comprehension in the human brain. *BioRxiv*.

Zhaomin Chen, Chai Kiat Yeo, Bu Sung Lee, and Chiew Tong Lau. 2018. Autoencoder-based network anomaly detection. In *2018 Wireless telecommunications symposium (WTS)*, pages 1–5. IEEE.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *MLCW*.

Hal Daumé III. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic. Association for Computational Linguistics.

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

Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. 2016. Multi30K: Multilingual English-German image descriptions. In *Proceedings of the 5th Workshop on Vision and Language*, pages 70–74, Berlin, Germany. Association for Computational Linguistics.

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.

John Giorgi, Osvald Nitski, Bo Wang, and Gary Bader. 2021. DeCLUTR: Deep contrastive learning for unsupervised textual representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 879–895, Online. Association for Computational Linguistics.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *ArXiv*, abs/2301.07597.

Md Akmal Haidar, Nithin Anchuri, Mehdi Rezagholizadeh, Abbas Ghaddar, Philippe Langlais, and Pascal Poupart. 2022. RAIL-KD: RAndom intermediate layer mapping for knowledge distillation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1389–1400, Seattle, United States. Association for Computational Linguistics.

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.

Dan Hendrycks, Mantas Mazeika, and Thomas G. Dietterich. 2019. Deep anomaly detection with outlier exposure. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of

language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Di Jin, Shuyang Gao, Seokhwan Kim, Yang Liu, and Dilek Hakkani-Tür. 2022. Towards textual out-of-domain detection without in-domain labels. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:1386–1395.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Machine Learning Proceedings 1995*, pages 331–339. Elsevier.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1311–1316, Hong Kong, China. Association for 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.

Bing Liu and Sahisnu Mazumder. 2021. Lifelong and continual learning dialogue systems: learning during conversation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 15058–15063.

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*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Tingting Ma, Qianhui Wu, Zhiwei Yu, Tiejun Zhao, and Chin-Yew Lin. 2022. On the effectiveness of sentence encoding for intent detection meta-learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3806–3818, Seattle, United States. Association for Computational Linguistics.

Kimberly T Mai, Toby Davies, and Lewis D Griffin. 2022. Self-supervised losses for one-class textual anomaly detection. *ArXiv preprint*, abs/2204.05695.

Andrei Manolache, Florin Brad, and Elena Burceanu. 2021. DATE: Detecting anomalies in text via self-supervision of transformers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 267–277, Online. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Gray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

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*, volume 35, pages 13675–13682.

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

Douglas A Reynolds. 2009. Gaussian mixture models. *Encyclopedia of biometrics*, 741(659-663).

Lukas Ruff, Yury Zemlyanskiy, Robert Vandermeulen, Thomas Schnake, and Marius Kloft. 2019. Self-attentive, multi-context one-class classification for unsupervised anomaly detection on text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4061–4071, Florence, Italy. Association for Computational Linguistics.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. 2020. Distributionally robust neural networks. In *8th International Conference on*

Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net.

Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. 2001. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471.

Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 2019. Cross-lingual transfer learning for multilingual task oriented dialog. 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 3795–3805, Minneapolis, Minnesota. Association for Computational Linguistics.

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*, pages 2911–2916, Copenhagen, Denmark. Association for Computational Linguistics.

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

Kaitao Song, Yichong Leng, Xu Tan, Yicheng Zou, Tao Qin, and Dongsheng Li. 2022. Transcormer: Transformer for sentence scoring with sliding language modeling. *ArXiv preprint*, abs/2205.12986.

David MJ Tax and Robert PW Duin. 2004. Support vector data description. *Machine learning*, 54(1):45–66.

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.

Hong Xu, Keqing He, Yuanmeng Yan, Sihong Liu, Zijun Liu, and Weiran Xu. 2020. A deep generative distance-based classifier for out-of-domain detection with mahalanobis space. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1452–1460, Barcelona, Spain (Online). International Committee on Computational Linguistics.

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 (Volume 1: Long Papers)*, pages 1052–1061, Online. Association for Computational Linguistics.

Eyup Halit Yilmaz and Cagri Toraman. 2020. KLOOS: KL divergence-based out-of-scope intent detection in human-to-machine conversations. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 2105–2108. ACM.

Houssam Zenati, Chuan Sheng Foo, Bruno Lecouat, Gaurav Manek, and Vijay Ramaseshan Chandrasekhar. 2018. Efficient gan-based anomaly detection. *ArXiv preprint*, abs/1802.06222.

Zhiyuan Zeng, Keqing He, Yuanmeng Yan, Zijun Liu, Yanan Wu, Hong Xu, Huixing Jiang, and Weiran Xu. 2021a. 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 and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 870–878, Online. Association for Computational Linguistics.

Zhiyuan Zeng, Keqing He, Yuanmeng Yan, Hong Xu, and Weiran Xu. 2021b. Adversarial self-supervised learning for out-of-domain detection. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5631–5639, Online. Association for Computational Linguistics.

Li-Ming Zhan, Haowen Liang, Bo Liu, Lu Fan, Xiao-Ming Wu, and Albert Y.S. Lam. 2021. Out-of-scope intent detection with self-supervision and discriminative training. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3521–3532, Online. Association for Computational Linguistics.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 649–657.

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*, pages 1100–1111, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Appendix

A.1 Dataset Details

- **Group 1: CLINC150.** Larson et al. (2019) introduce a crowdsourced dialog dataset. Following Xu et al. (2021), we use all training queries covering 150 intents as ID training data and fuse 4500 ID examples of the test split with 1000 OoD examples for evaluation.

- **Group 2: SST.** Following Hendrycks et al. (2020) and Xu et al. (2021), we use the training split of the SST dataset (Socher et al., 2013) for ID training examples and use its test split as ID test examples. The same described random sample of 500 examples from 20 NewsGroups (Lang, 1995), English-German Nulti30K (Elliott et al., 2016), RTE (Dagan et al., 2005), and SNLI (Bowman et al., 2015), is combined and used as OoD test examples.

- **Group 3: ROSTD.** Gangal et al. (2020) releases a dataset consisting of 4590 OoD examples with respect to the English split of Schuster et al. (2019) as the ID dataset. Here we use the training ID, test-ID, and actual OoD as in Gangal et al. (2020) for fair comparison.

- **Group 4: 20NewsGroups.** Following Manolache et al. (2021), we only consider articles from six top-level classes of 20NewsGroups (Lang, 1995) for evaluation. We construct the ID data using examples from a single label, *i.e.*, training split for ID training and test split for ID test. We take data corresponding to other labels in the test split as OoD test examples.

- **Group 5: AGNews.** AG News is a topic classification dataset (Zhang et al., 2015) collected from various news sources. There are four topics in total. Similar to Group 4, we conduct experiments with each single label for ID and others for OoD, respectively.

- **Group 6: HC3.** Human ChatGPT Comparison Corpus (HC3) is a question-answer dataset (Guo et al., 2023), which collects the human (from wiki and public QA datasets) and ChatGPT answers for the same questions. HC3 covers a wide range of domains (open-domain, computer science, finance, medicine, law, and psychology). We conduct experiments with ChatGPT answers for ID and human answers for OoD at the sentence level, respectively.

Table 7 shows the statistics of the different datasets and sub-topics, if any.

Group		# of ID (train)	# of ID (test)	# of OoD (test)
#1: CLINC150		15000	4500	1000
#2: SST		8544	2210	2000
#3: ROSTD		30521	8621	4590
#4: 20NewsGroups	comp	2857	1909	5390
	misc	577	382	6917
	pol	1531	1025	6274
	rec	2301	1524	5775
	rel	1419	939	6360
	sci	2311	1520	5779
#5: AGNews	business	30000	1900	5700
	sci	30000	1900	5700
	sports	30000	1900	5700
	world	30000	1900	5700
#6: HC3		13442	13443	58546

Table 7: Dataset statistics.

A.2 *MDF+IMLM* with Different Base Models

We take MDF+IMLM from Xu et al. (2021) as one of the baselines. In the main body of this paper, we show the results of MDF+IMLM with BERT as the base model because BERT is the most considered counterpart for GPT-2-small used in our approach. Here we include the RoBERTa-based results of MDF+IMLM from Xu et al. (2021) for your information.

Table 8 shows that using a more powerful base model does bring significant performance gain to MDF+IMLM. Though our model is implemented with GPT-2-small, it still demonstrates comparable (on SST) and even superior performance (CLINC150) with RoBERTa based MDF+IMLM.

A.3 Discussion on CLM and MLM.

Here we discuss the consideration for using CLM rather than MLM. In fact, we conducted experiments using the previous method of masking X% of tokens for one forward. However, the results were not satisfactory. We attribute this to an insufficient perplexity estimation in a single forward. In other words, with MLM, it would be better to recover the joint probability of the entire input sequence to

Method	CLINC150			SST		
	AUROC (↑)	AUPR (↑)	FAR95 (↓)	AUROC (↑)	AUPR (↑)	FAR95 (↓)
MDF+IMLM (GPT-2-small)‡	72.03	31.30	74.29	-	-	-
MDF+IMLM (BERT)†	77.8	39.1	-	93.6	89.4	-
MDF+IMLM (BERT)‡	77.46	39.23	65.87	96.79	95.62	11.68
MDF+IMLM (RoBERTa)†	80.1	44.9	-	**99.9**	**99.8**	-
Ours (GPT-2-small)	**92.51**	**70.94**	**27.16**	97.97	97.81	**9.50**

Table 8: Performance comparison on CLINC150 and SST. † represents results reported in Xu et al. (2021). ‡ denotes our re-implemented results.

achieve better performance, *i.e.*, forwarding an input sentence multiple times so that the probability of each token in the sentence could be predicted. This should be time-consuming and thus we use CLM in this paper.

ACL 2023 Responsible NLP Checklist

A For every submission:

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

☑ A2. Did you discuss any potential risks of your work?
"Limitations"

☑ A3. Do the abstract and introduction summarize the paper's main claims?
"Abstraction", "Introduction"

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

B ☑ Did you use or create scientific artifacts?

"Implementation details"

☐ B1. Did you cite the creators of artifacts you used?
No response.

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

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

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
We only use open-source benchmark datasets.

☐ 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.
"Experiments"

C ☑ Did you run computational experiments?

"Experiments"

☒ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
GPT2-small,124M, 1x V100 16G, 60min.

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?
"implementation details"

☑ 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?
"Experiments"

☑ 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.)?
"Experiments"

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

We only use public benchmark datasets.

☐ 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.