Towards Textual Out-of-Domain Detection without any In-Domain Labels

Di Jin, Shuyang Gao, Seokhwan Kim, Yang Liu, Dilek Hakkani-Tur
Amazon Alexa AI
Sunnvayle, CA 94089
{djinamzn,shuyag,seokhwk,yangliud,hakkanit}@amazon.com

Abstract

In many real-world settings, machine learning models need to identify user inputs that are out-of-domain (OOD) so as to avoid performing wrong actions. This work focuses on a challenging case of OOD detection, where no labels for in-domain data are accessible (e.g., no intent labels for the intent classification task). To this end, we propose a novel representation learning based method by combining unsupervised clustering and contrastive learning so that better data representations for OOD detection can be learned. Through extensive experiments, we demonstrate that this method can be even competitive to the state-of-the-art supervised approaches with label information.

1 Introduction

Deep learning models are widely used in many real-life applications, and for many classification problems. At test time, models need to identify examples that differ significantly from the model's training data distribution, i.e., out-of-domain detection. For example, in dialog agents such as Amazon Alexa and Google Home Assistant, detecting unknown or out-of-domain (OOD) intents from user queries is an essential component in order to know when a query falls outside their range of predefined, supported intents [29]. Correctly identifying out-of-scope cases is especially crucial in deployed systems, both to avoid performing the wrong action and also to identify potential future directions for development. However, for state-of-the-art deep learning classifiers, it has been widely observed that their raw probability values are often over-calibrated, i.e., it has high values even for OOD inputs [8, 9]. This necessitates having a specially designed mechanism for OOD detection.

The task of OOD detection has historically been explored in related forms under various names such as outlier detection, anomaly detection, open classification, etc. [9, 1, 25, 11, 12]. Most of previous work relied on the existing class labels (ID labels hereafter) for in-domain multi-class classification tasks. For example, for OOD detection for the intent classification task, current work implements OOD detection based on a classifier that has been well trained using the intent labels for ID data [29, 18, 33]. In contrast, our work addresses the OOD detection problem in the scenario where no ID labels are available (e.g., no intent labels for the multi-class intent classification task) and thus most existing solutions are not directly applicable.

Current available methods for tackling this *OOD detection without ID labels* setting can be categorized into two threads: language model (LM) likelihood based methods [20, 6], and representation learning based one-class classification method [22, 26]. For the first thread, [3, 16] find that likelihood is poor at separating out OOD examples; in some cases even assigning them higher likelihood than the ID test split. [20] made similar observations for detecting OOD DNA sequences and thus proposed to "correct" the original likelihood with one from a "background" model trained on noisy inputs, termed as likelihood ratio (LR). [6] explored the use of LR method on NLP tasks and reported good performance. For the second thread, we can treat all of the ID data as one-class data and those OOD data as other classes, forming a one-class classification problem. [26] proposed to first learn a good

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

representation model via contrastive learning and then learn a density estimator on the obtained representations, which has shown state-of-the-art performance for one-class classification.

In this work, we propose a novel method for the second thread of representation learning based methods. We find that directly applying representation learning based one-class classification methods to our problem cannot yield satisfying performance, since ID data indeed consists of multiple classes that we do not have access to. To solve this issue, we propose to perform both unsupervised clustering and contrastive learning when we learn representations of ID data, which can help produce sharper boundaries between ID and OOD samples. To the best of our knowledge, we are the first to apply the representation learning based one-class classification method to OOD detection in NLP tasks and significantly improve it so that it can rival likelihood based methods. Our proposed methods for textual OOD detection without ID labels have achieved very impressive performance, even close to those achieved by supervised methods that use ID labels for training for three out of four datasets.

2 Related Works

All OOD detection methods for unsupervised detection rely on producing a score that can be compared with a threshold to differentiate between ID and OOD samples. One major line of methods utilize classifiers trained on the ID data to calculate the **probability** of the test instance and use it as comparison score. The maximum softmax probability is recognized as a strong baseline [9]. [13] found that increasing the softmax temperature τ makes the resulting probability more discriminative for OOD Detection. [11] and [34] utilized ID inputs and unlabeled data to generate pseudo-OOD utterances around the decision boundaries with a generative adversarial network (GAN) so as to calibrate the output probabilities. Another important line of methods use **distance estimation** as the OOD score: [12] proposed using Mahalanobis distances to per-class Gaussians in the intermediate representation learned by the classifier. Specifically, a Gaussian distribution is fit for each training class from all training points in that class. Following this work, [10] analyzed the effectiveness of Mahalanobis distance and proposed several variants. [30] and [18] applied it to NLP tasks and reported strong performance.

All of the above previous work assumed the availability of well-trained classifiers on ID data with annotated ID labels. However, in some cases, such annotated labels for classification may not be available, posing a great challenge to OOD detection. Our work will focus on addressing this special case of **OOD detection without ID labels**. It has been rarely explored previously. The only available ready-to-use method is the likelihood ratio method, where two deep generative models are trained on normal and noisy inputs, respectively, and then the difference of likelihood produced by these two models is used as the OOD score [20, 6, 18]. In this study, we compare that work with other ways of building the background model and evaluate the effect of length normalization for likelihood scores. We find that a generic pre-trained language model without domain adaptation is a better choice for the background model.

3 Methods

In this section, we would like to propose a representation learning based method for OOD detection without any ID labels. More specifically, the representation learning is implemented via a classic unsupervised method: unsupervised clustering. Combined with a recent popular technique, contrastive learning, it can significantly boost the OOD detection performance.

Our method is mainly composed of two steps: adaptive representation learning and density estimation.

- Adaptive representation learning: In this step, we learn the ID data distribution with a pre-trained sentence encoder ψ by optimizing the following overall objective:

$$\mathcal{L} = \mathcal{L}_{cluster} + \gamma \mathcal{L}_{CL}, \tag{1}$$

  where $\mathcal{L}_{cluster}$ denotes the unsupervised clustering objective and $\mathcal{L}_{CL}$ denotes the contrastive learning objective, both of which are described in more details below, and γ is tuned on the validation set.

- Density estimation: In this step, we feed each ID sentence $\mathbf{x}_i^{ID}$ into the encoder ψ and obtain the mean vector of all the tokens' hidden states from the last layer, which is used as

its representation vector: $\psi(\mathbf{x}_i^{ID})$. Then we learn a density estimator D over these vectors, such as One-class Support Vector Machine (OC-SVM), Kernel Density Estimation (KDE), and Gaussian Mixture Model (GMM) [21]. We choose GMM in our work since we find it consistently outperforms other choices in experiments.

During inference, given a test sample $\mathbf{x}$, we first encode it with ψ and then use the density estimator D to produce a density score $D(\psi(\mathbf{x}))$. If this score is above a pre-set threshold η, this sample is considered as an ID sample, otherwise as an OOD sample.

3.1 Clustering

Our ID data may contain multiple classes, but we do not have such annotations. Therefore, we would like to adopt unsupervised clustering to learn this implicit prior. Suppose our ID data consists of K semantic categories, and each category is characterized by its centroid in the representation space, denoted as $\mu_k, k \in \{1, ..., K\}$. Here μ_k is iteratively refined during the training phase. Note that since we do not know how many clusters there are for the ID data, K is treated as a hyper-parameter and tuned using the validation set. Overall, the objective for unsupervised clustering is to push the cluster assignment probability q_i for input text x_i towards the target distribution p_i, which is achieved by optimizing the KL divergence between them:

$$\ell_i^C = KL(p_i||q_i) = \sum_{k=1}^{K} p_{ik} \log \frac{p_{ik}}{q_{ik}},$$

where q_{ik} and p_{ik} are the predicted and target probability of assigning x_i to the k^{th} cluster, respectively. The overall clustering objective is then defined as:

$$\mathcal{L}_{cluster} = \frac{1}{M} \sum_{i=1}^{M} \ell_i^C, \tag{2}$$

where M is the batch size.

Following [27], we use the Student's t-distribution to compute q_{ik} as below:

$$q_{ik} = \frac{(1+ \parallel e_i - \mu_k \parallel_2^2 / \alpha)^{-\frac{\alpha+1}{2}}}{\sum_{k'=1}^{K} (1+ \parallel e_i - \mu_k' \parallel_2^2 / \alpha)^{-\frac{\alpha+1}{2}}},$$

where $e_i = \psi(x_i)$, and α denotes the degree of freedom of the Student's t-distribution, which is set as 1 in this work.

Since we do not have the ground truth of p_{ik}, we approximate it following [28]:

$$p_{ik} = \frac{q_{ik}^2 / f_k}{\sum_{k'=1}^{K} q_{ik'}^2 / f_{k'}},$$

where $f_k = \sum_{i=1}^{M} q_{ik}$. This target distribution first sharpens the soft-assignment probability q_{ik} by raising it to the second power, and then normalizes it by the associated cluster frequency. By doing so, we encourage learning from high confidence cluster assignments and simultaneously combating the bias caused by imbalanced clusters.

3.2 Contrastive Learning

Following [32], we add a contrastive learning training objective [2] while performing clustering, which can help stabilize and improve the clustering process. For a randomly sampled mini-batch $\mathcal{B} = \{x_{i^0}\}_{i^0=1}^{M}$ (all come from ID samples), we randomly generate an augmentation x_{i^1} for each data instance x_{i^0} so that x_{i^0} and x_{i^1} form a pair of positive instances, yielding an augmented mini-batch $\mathcal{B}^a$ of size $2M$. Following [7], to obtain the data augmentation, we pass the same input sentence to the pre-trained sentence encoder twice and obtain two embeddings as "positive pairs", by applying independently sampled dropout masks (current prevalent pre-trained encoders are all based on transformer architecture and contain dropout masks in each layer). Although strikingly simple, this approach has been found to outperform many other complex text augmentation methods, such as contextual word replacement and back-translation [7]. To achieve contrastive learning, for x_{i^0} we try

to bring it close to its positive counterpart x_{i^1} while moving it far away from other instances in the mini-batch $\mathcal{B}^a$, which is implemented by minimizing:

$$\ell_{i^0}^{CL} = -\log \frac{\exp(sim(z_{i^0}, z_{i^1})/\tau)}{\sum_{j=1}^{2M} \mathbb{1}_{j \neq i^0} \cdot \exp(sim(z_{i^0}, z_j)/\tau)},$$

where $z_j = g(\psi(x_j))$ and g is implemented by a two-layers full connected network in this work; τ denotes the temperature parameter which we set as 0.5; $sim(z_i, z_j) = z_i^T z_j / ||z_i||_2 ||z_j||_2$ following [2]. Then the overall contrastive learning objective is defined as:

$$\mathcal{L}_{CL} = \frac{1}{2M} \sum_{i=1}^{2M} \ell_i^{CL}. \tag{3}$$

4 Experiments

4.1 Datasets

We have experimented with four text classification datasets: two on intent classification for dialogue systems, one on question topic classification, and one on search snippets topic classification. Data statistics are in the Appendix.

SNIPS consists of user utterances for 7 intent classes such as GetWeather, RateBook, etc. [4]. Since this dataset itself does not include OOD utterances, we follow the procedure described in [14] to synthetically create OOD examples. Intent classes covering at least 75% of the training points in combination are retained as ID. Examples from the remaining classes are treated as OOD and removed from the training set. In the validation and test sets, examples from these classes are relabelled to the single class label OOD. Since multiple ID-OOD splits of the classes satisfying these ratios are possible, our results are averaged across 5 randomly chosen splits.

ROSTD starts from the English part of multilingual dialog dataset released by [24] as ID utterances and later gets extended by [6] with OOD utterances.

Stackoverflow is a subset of the challenge data published by Kaggle, where question titles associated with 20 different categories are selected by [31]. We follow the same way as SNIPS to create OOD samples. We report average results across 5 randomly chosen splits.

Searchsnippets is extracted from web search snippets, which contains search snippets associated with 8 different topics [17]. We randomly selected 2 out of 8 classes as OOD classes while using the remaining classes as ID classes. Again we report average results across 5 randomly chosen splits.

4.2 Evaluation Metrics

Following [9], we use the following metrics to measure OOD detection performance:

$\mathbf{FPR@95\%TPR}$ corresponds to False Positive Ratio with the decision threshold being set to $\theta = sup\{\tilde{\theta} \in \mathcal{R} | TPR(\tilde{\theta}) \leq 95\%\}$, where TPR is the True Positive Ratio. Note here the OOD class is the positive class.

$\mathbf{AUROC}$ measures the area under the Receiver Operating Characteristic, also known as the ROC curve. Note that this curve is for the OOD class.

$\mathbf{AUPR_{OOD}}$ measures the area under Precision-Recall Curve, taking OOD as the positive class. It is more suitable for highly imbalanced data in comparison to $AUROC$.

4.3 Experimental Settings

The sentence encoder ψ we used is DistilBERT-Base-NLI (denoted also as DistilBERT in the rest of this document), which is obtained by fine-tuning DistilBERT-Base on MultiNLI and SNLI datasets [19]. By tuning on the validation set, we set the number of components for GMM as 1 and set γ as 1.0, 0.1, 1.0, and 4.0 for ROSTD, SNIPS, Stackoverflow, and Searchsnippets, respectively.

We compare our representation learning based method with the following baselines:

Table 1: Comparison of OOD detection performance for the representation learning based methods. The bold font highlights the best results achieved among methods without using ID labels. † denotes that the best result is achieved by the RoBERTa-Large-NLI-SimCSE encoder while ‡ represents it is from DistilBERT-Base-NLI. "CL" denotes "contrastive learning".

Dataset	ID Labels	Model	$\mathbf{AUROC}(\%)\uparrow$	$\mathbf{AUPR_{OOD}}(\%)\uparrow$	$\mathbf{FPR@95\%TPR}(\%)\downarrow$
ROSTD	NO	Encoder w/o Fine-tuning†	**99.77**	**99.46**	**0.71**
		Fine-tuned Encoder via MLM†	99.29	98.11	3.00
		Ours	<u>99.73</u>	<u>99.30</u>	<u>0.73</u>
		$-\mathcal{L}_{cluster}$	97.78	93.91	9.58
		$-\mathcal{L}_{CL}$	98.88	96.45	4.87
	YES	DistilBERT Maha	99.61	98.85	1.60
		RoBERTa Maha	99.80	99.50	0.50
SNIPS	NO	Encoder w/o Fine-tuning†	94.09 ± 3.07	82.31 ± 10.04	17.78 ± 7.08
		Fine-tuned Encoder via MLM‡	94.76 ± 3.31	86.53 ± 8.07	21.54 ± 15.03
		Ours	$\mathbf{97.56} \pm 1.01$	$\mathbf{92.80} \pm 3.19$	$\mathbf{8.53} \pm 4.42$
		$-\mathcal{L}_{cluster}$	84.54 ± 8.95	64.02 ± 18.70	43.57 ± 19.08
		$-\mathcal{L}_{CL}$	97.17 ± 1.37	92.22 ± 3.97	15.75 ± 8.93
	YES	DistilBERT Maha	97.93 ± 1.37	95.29 ± 3.03	8.65 ± 6.42
		RoBERTa Maha	98.30 ± 0.98	95.10 ± 3.80	7.06 ± 3.51
Stackoverflow	NO	Encoder w/o Fine-tuning‡	73.07 ± 3.77	55.69 ± 3.40	76.54 ± 1.62
		Fine-tuned Encoder via MLM‡	78.59 ± 1.68	62.18 ± 1.63	64.13 ± 5.04
		Ours	$\mathbf{83.65} \pm 0.13$	$\mathbf{62.22} \pm 2.85$	$\mathbf{36.82} \pm 1.84$
		$-\mathcal{L}_{cluster}$	70.99 ± 1.99	53.76 ± 3.00	78.43 ± 3.32
		$-\mathcal{L}_{CL}$	50.79 ± 13.13	37.99 ± 13.54	90.80 ± 5.24
	YES	DistilBERT Maha	92.00 ± 0.96	78.35 ± 3.24	19.99 ± 3.02
		RoBERTa Maha	91.98 ± 1.34	76.55 ± 5.62	19.61 ± 2.51
Searchsnippets	NO	Encoder w/o Fine-tuning†	82.93 ± 5.05	88.22 ± 3.78	61.80 ± 11.02
		Fine-tuned Encoder via MLM†	84.32 ± 1.96	89.27 ± 2.15	62.54 ± 4.24
		Ours	$\mathbf{94.70} \pm 2.29$	$\mathbf{96.16} \pm 1.92$	$\mathbf{22.43} \pm 7.20$
		$-\mathcal{L}_{cluster}$	88.09 ± 6.63	90.91 ± 5.73	41.09 ± 17.76
		$-\mathcal{L}_{CL}$	78.95 ± 4.93	82.01 ± 4.22	65.06 ± 11.53
	YES	DistilBERT Maha	95.09 ± 2.43	96.88 ± 1.56	25.13 ± 14.00
		RoBERTa Maha	97.26 ± 1.05	98.38 ± 0.64	14.61 ± 7.00

Encoder w/o Fine-tuning We directly use a pre-trained encoder ψ as an off-the-shelf model without any fine-tuning. Specifically, we used DistilBERT as well as the RoBERTa-Large-NLI-SimCSE (denoted also as RoBERTa later) that is obtained by fine-tuning RoBERTa-Large on NLI datasets via contrastive learning and is the state-of-the-art encoder for sentence embedding [7].[1] We report the best result between these two encoders.

Fine-tuned Encoder via MLM We fine-tune DistilBERT and RoBERTa on ID data via masked language modeling (MLM) [5] to obtain domain data adapted ψ and report the best result.

DistilBERT/RoBERTa Maha We re-implemented a supervised method from [18] by training a DistilBERT and a RoBERTa model on ID data with those ID labels and then performing OOD detection via Mahalanobis distance, which is the state-of-the-art method for textual OOD detection.

$-\mathcal{L}_{cluster}$ **&** $-\mathcal{L}_{CL}$ Either unsupervised clustering by optimizing Equation 2 or contrastive learning by optimizing Equation 3 can act as strong baselines since they can both individually serve as approaches for unsupervised representation learning. We also compare with these two baselines in the ablation study.

5 Results

Table 1 summarizes the main results of OOD detection for the representation learning based methods. From this table, we have the following observations:

- Simply applying a generic pre-trained encoder without any fine-tuning on the ID data to obtain data representations and then performing density estimation can already achieve good performance (refer to Encoder w/o Fine-tuning).
- Adapting the pre-trained encoder on the ID data via MLM, i.e., fine-tuned encoder via MLM, shows some improvement for most datasets and metrics, e.g., on SNIPS, Stackoverflow, and Searchsnippets datasets for the $AUROC$ and $AUPR_{OOD}$ metrics.

[1] https://github.com/princeton-nlp/SimCSE

- Our proposed method for adapting the encoder via clustering and contrastive learning can significantly boost the OOD detection performance consistently for all datasets, outperforming all baselines.
- Our method can be comparable to the state-of-the-art method that intensively utilizes the ID labels for supervised training (i.e., DistilBERT/RoBERTa Maha) for ROSTD, SNIPS, and Searchsnippets datasets.

It is worth pointing out that our method is based on a small-size encoder, i.e., DistilBERT-Base (around 66M parameters), however, those baselines including Encoder w/o Fine-tuning, Finetuned Encoder, and RoBERTa Maha have adopted an encoder of much larger size, i.e., RoBERTa-Large with around 355M parameters. Moreover, the RoBERTa-Large encoder has been pre-trained on around 10 times larger corpora compared with DistilBERT-Base, thus achieving much better performance on various kinds of language understanding tasks, e.g., the average score of 88.5 achieved by RoBERTa-Large vs 77.0 achieved by DistilBERT-Base on the development sets of GLUE benchmark [23, 15]. Although our method is equipped with an encoder with much smaller model size (about 1/5) and inferior down-stream tasks performance, it can still beat those baselines using larger models when no ID labels are used, and be on par with the baseline that does use ID labels. This comparison demonstrates that our method can achieve both high OOD detection performance and parameter & computation efficiency, which fulfills the two consideratas for real-world deployment in industry applications.

6 Discussions

6.1 Ablation Study

We did an ablation study by removing either $\mathcal{L}_{cluster}$ or $\mathcal{L}_{CL}$ when optimizing Equation 1 for the representation learning based methods. These results are included in Table 1. By comparing with the full solution, we see that: (1) Both training objectives contribute to the improvement. (2) In some cases (e.g., Stackoverflow and Searchsnippets datasets), removing $\mathcal{L}_{CL}$ would cause larger performance degradation, while in the other cases, $\mathcal{L}_{cluster}$ is more important. This is dependent on the distributions of the ID samples (separation of different classes) and how OOD data is separated from ID data. We have provided a detailed analysis over representations we have learned via visualization to help compare different settings in the Appendix.

6.2 Influence of K

Since we do not have any information on the ID labels, we need to tune the number of clusters K on the validation set and find the optimal one when we perform unsupervised clustering. Table 2 shows these values for different data sets along with their numbers of classes for the ID data for the representation learning based methods. We can see that the optimal number of clusters for OOD detection is not equal to the actual number of labels within ID data. For example, the optimal K is 30 for the Searchsnippets dataset, which is far away from the number of labels, i.e., 6. This implies that our method is not simply an unsupervised approach to learning the classification task but rather a method more suitable for learning ID data distribution.

Table 2: Comparison between the optimal number of clusters and the number of ID labels. Since three datasets have 5 random splits, we list both minimum and maximum number of labels within the parentheses.

	ROSTD	**SNIPS**	**Stackoverflow**	**Searchsnippets**
Optimal K	14	6	17	30
Num of labels	12	(5, 5)	(13, 14)	(6, 6)

7 Conclusion

In this work, we aim at the OOD detection problem without using any ID labels. In addition to evaluating different LM based likelihood methods, we propose a representation learning based method by performing unsupervised clustering and contrastive learning to learn good data representations for OOD detection. We demonstrate that this novel unsupervised method can not only outperform the best likelihood based methods but also be even competitive to the state-of-the-art supervised method that has extensively used labeled data.

References

[1] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 93–104, 2000.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[3] Hyunsun Choi and Eric Jang. Generative ensembles for robust anomaly detection, 2019.

[4] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*, 2018.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, June 2019. Association for Computational Linguistics.

[6] Varun Gangal, Abhinav Arora, Arash Einolghozati, and Sonal Gupta. Likelihood ratios and generative classifiers for unsupervised out-of-domain detection in task oriented dialog. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7764–7771, 2020.

[7] Tianyu Gao, Xingcheng Yao, and Danqi Chen. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*, 2021.

[8] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017.

[9] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations*, 2017.

[10] Ryo Kamoi and Kei Kobayashi. Why is the mahalanobis distance effective for anomaly detection? *arXiv preprint arXiv:2003.00402*, 2020.

[11] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations*, 2018.

[12] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[13] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018.

[14] Ting-En Lin and Hua Xu. 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, July 2019. Association for Computational Linguistics.

[15] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019.

[16] Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *International Conference on Learning Representations*, 2019.

[17] Xuan-Hieu Phan, Le-Minh Nguyen, and Susumu Horiguchi. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th international conference on World Wide Web*, pages 91–100, 2008.

[18] Alexander Podolskiy, Dmitry Lipin, Andrey Bout, Ekaterina Artemova, and Irina Piontkovskaya. Revisiting mahalanobis distance for transformer-based out-of-domain detection. *arXiv preprint arXiv:2101.03778*, 2021.

[19] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019.

[20] Jie Ren, Peter J. Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[21] Douglas Reynolds. *Gaussian Mixture Models*, pages 659–663. Springer US, Boston, MA, 2009.

[22] Lukas Ruff, Yury Zemlyanskiy, Robert Vandermeulen, Thomas Schnake, and Marius Kloft. 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, July 2019. Association for Computational Linguistics.

[23] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108, 2019.

[24] Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 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, June 2019. Association for Computational Linguistics.

[25] Lei Shu, Hu Xu, and Bing Liu. 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, September 2017. Association for Computational Linguistics.

[26] Kihyuk Sohn, Chun-Liang Li, Jinsung Yoon, Minho Jin, and Tomas Pfister. Learning and evaluating representations for deep one-class classification. In *International Conference on Learning Representations*, 2021.

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

[28] Junyuan Xie, Ross B. Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *ICML*, pages 478–487, 2016.

[29] Hong Xu, Keqing He, Yuanmeng Yan, Sihong Liu, Zijun Liu, and Weiran Xu. 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, 2020.

[30] Hong Xu, Keqing He, Yuanmeng Yan, Sihong Liu, Zijun Liu, and Weiran Xu. 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), December 2020. International Committee on Computational Linguistics.

[31] Jiaming Xu, Bo Xu, Peng Wang, Suncong Zheng, Guanhua Tian, and Jun Zhao. Self-taught convolutional neural networks for short text clustering. *Neural Networks*, 88:22–31, 2017.

[32] Dejiao Zhang, Feng Nan, Xiaokai Wei, Shangwen Li, Henghui Zhu, Kathleen McKeown, Ramesh Nallapati, Andrew Arnold, and Bing Xiang. Supporting clustering with contrastive learning. *arXiv preprint arXiv:2103.12953*, 2021.

[33] Jianguo Zhang, Kazuma Hashimoto, Wenhao Liu, Chien-Sheng Wu, Yao Wan, Philip Yu, Richard Socher, and Caiming Xiong. Discriminative nearest neighbor few-shot intent detection by transferring natural language inference. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5064–5082, Online, November 2020. Association for Computational Linguistics.

[34] Yinhe Zheng, Guanyi Chen, and Minlie Huang. Out-of-domain detection for natural language understanding in dialog systems. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1198–1209, 2020.

A Data

Table A.1 summarizes the statistics of these four datasets. To be noted, the labels in these datasets are only used for splitting ID and OOD samples and not used for training the OOD detector.

Table A.1: Dataset statistics. Except ROSTD, all numbers are averaged across 5 chosen splits.

Statistic	**ROSTD**	**SNIPS**	**Stackoverflow**	**Searchsnippets**
Train-ID	30,521	9,332	10,020	9,163
Valid-ID	4,181	500	428	1,238
Valid-OOD	1,500	200	229	1,292
Test-ID	8,621	506	418	1,237
Test-OOD	3,090	193	235	1,293

B Visualization of Representations

To obtain a qualitative sense of how our proposed representation learning method works, we provide T-SNE visualization plots of sentence representations for both ID and OOD samples of the Searchsnippets, SNIPS, Stackoverflow, and ROSTD datasets in Figure B.1, B.2, B.3, and B.4, respectively. As can be seen, the unsupervised clustering module can aggregate data points into many clusters by bringing close sentences of the same classes while pushing away sentences of different classes (e.g., Figure B.1B, B.2B, B.3B, and B.4B). In contrast, the contrastive learning module can distribute data points uniformly over the whole latent space, in which process ID and OOD samples can be pushed away from each other (e.g., Figure B.1C). By combining these two modules, we can realize both goals and make the boundaries between ID and OOD samples much more clear compared with no adaptive fine-tuning (e.g., Figure B.1A vs. Figure B.1D), and thus achieve the optimal OOD detection performance. Note when both modules are combined, the OOD samples tend to be condensed into the center, while ID samples are scattered outside of OOD samples in clusters (e.g., Figure B.1D, B.2D, B.3D, and B.4D), which facilitates the density estimator to better differentiate ID and OOD samples.

More specifically, for the Searchsnippets dataset, without the help of contrastive learning, representations obtained via only clustering still cannot well differentiate OOD samples from ID samples because ID and OOD samples still mingle with each other for some clusters (see Figure B.1B). In contrast, from Figure B.1C, we find that contrastive learning alone can better separate ID and OOD samples than no adaptation by condensing the ID data distribution (red dots) and spreading OOD samples outside (blue dots). In this case, the contrastive learning module contributes more to the full model performance than the clustering module due to its better capability at differentiating ID and OOD samples, which aligns with the conclusion drawn from the ablation study in Table 1. However, for the ROSTD and SNIPS datasets, unsupervised clustering itself can already well separate ID and OOD samples by distributing the ID samples into many condensed clusters, where OOD samples are distributed outside of the ID clusters. This explains the finding revealed by the Ablation study in Section 6.1 that the clustering objective contributes more to the full model performance.

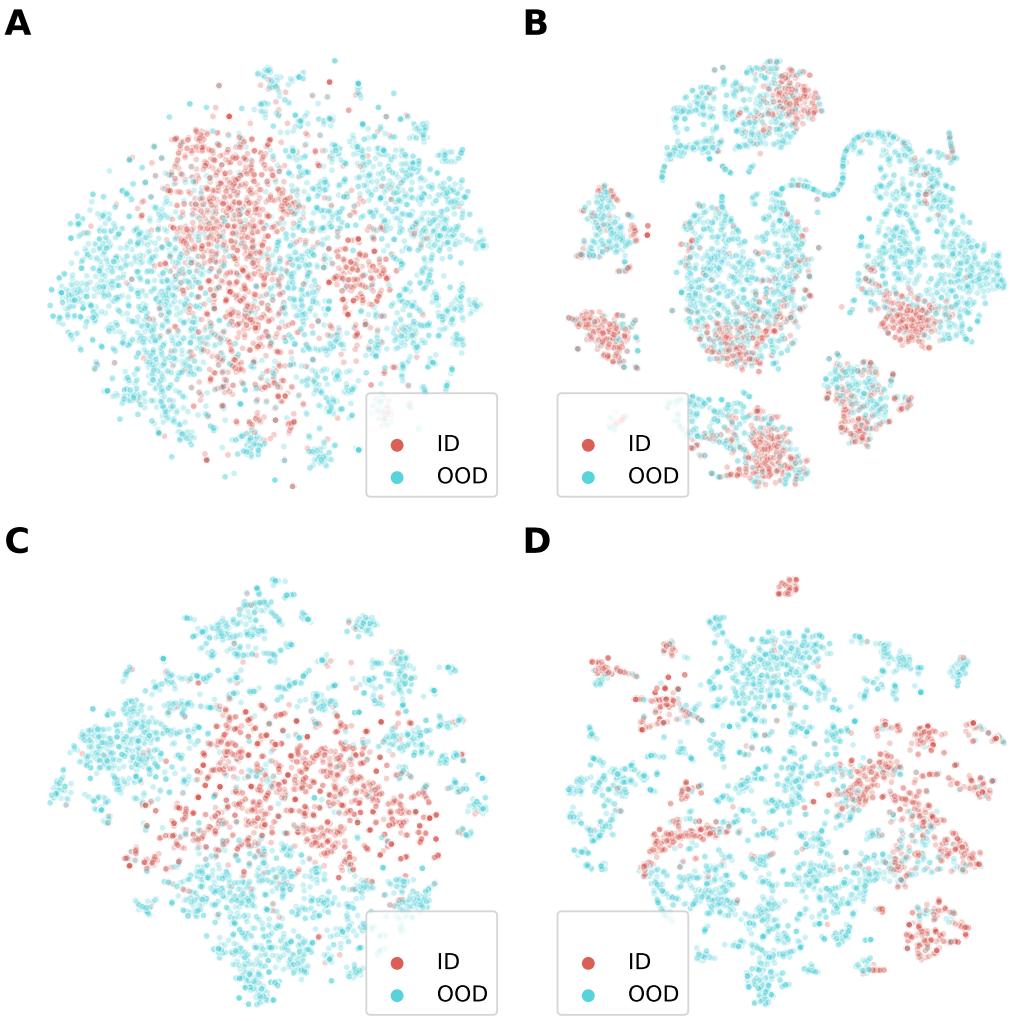

Figure B.1: T-SNE visualization of learned sentence representations for both ID and OOD samples of Searchsnippets dataset. A: DistilBERT-Base-NLI; B: DistilBERT-Base-NLI fine-tuned via unsupervised clustering; C: DistilBERT-Base-NLI fine-tuned via contrastive learning; D: DistilBERT-Base-NLI fine-tuned via combining clustering and contrastive learning.

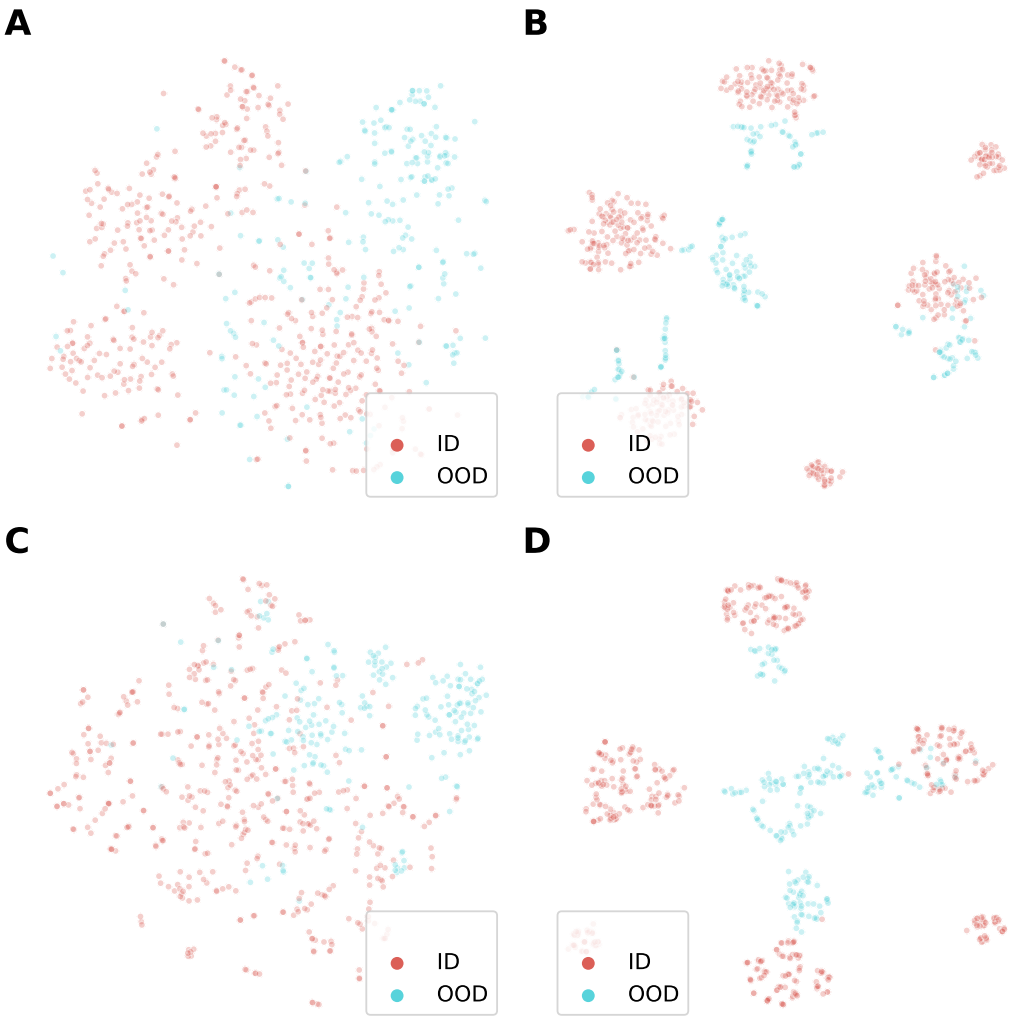

Figure B.2: T-SNE visualization of learned sentence representations for both ID and OOD samples of SNIPS dataset. A: DistilBERT-Base-NLI; B: DistilBERT-Base-NLI fine-tuned via unsupervised clustering; C: DistilBERT-Base-NLI fine-tuned via contrastive learning; D: DistilBERT-Base-NLI fine-tuned via combining clustering and contrastive learning.

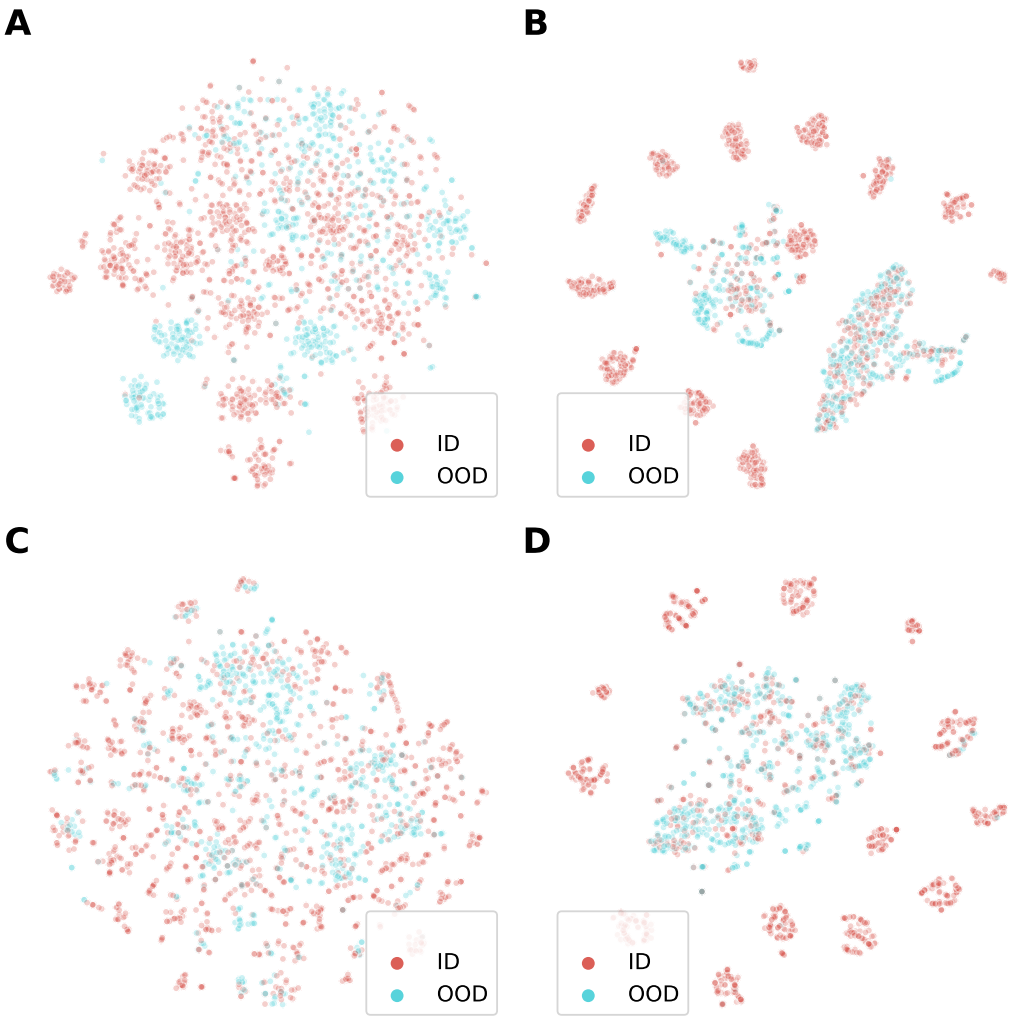

Figure B.3: T-SNE visualization of learned sentence representations for both ID and OOD samples of Stackoverflow dataset. A: DistilBERT-Base-NLI; B: DistilBERT-Base-NLI fine-tuned via unsupervised clustering; C: DistilBERT-Base-NLI fine-tuned via contrastive learning; D: DistilBERT-Base-NLI fine-tuned via combining clustering and contrastive learning.

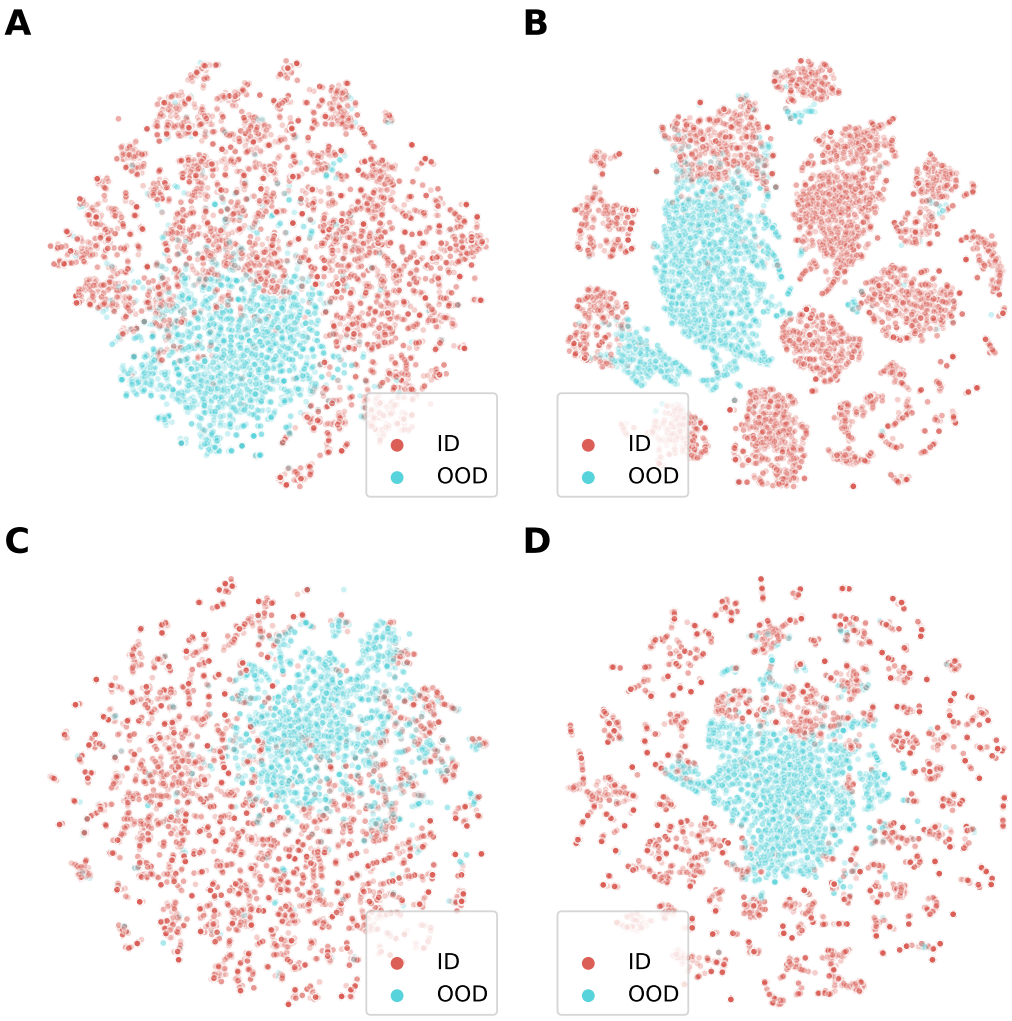

Figure B.4: T-SNE visualization of learned sentence representations for both ID and OOD samples of ROSTD dataset. A: DistilBERT-Base-NLI; B: DistilBERT-Base-NLI fine-tuned via unsupervised clustering; C: DistilBERT-Base-NLI fine-tuned via contrastive learning; D: DistilBERT-Base-NLI fine-tuned via combining clustering and contrastive learning.