Improving Unsupervised Out-of-domain Detection through Pseudo Labeling and Learning

Byounghan Lee[1] **Jaesik Kim**[1,2] **Junekyu Park**[1,3] **Kyung-Ah Sohn**[1,*]
Ajou University[1], University of Pennsylvania[2], Superb AI[3]
{qudgks96, kasohn}@ajou.ac.kr
Jaesik.Kim@pennmedicine.upenn.edu
idbluefish@gmail.com

Abstract

Unsupervised out-of-domain (OOD) detection is a task aimed at discriminating whether given samples are from the in-domain or not, without the categorical labels of in-domain instances. Unlike supervised OOD, as there are no labels for training a classifier, previous works on unsupervised OOD detection adopted the one-class classification (OCC) approach, assuming that the training samples come from a single domain. However, in-domain instances in many real world applications can have a heterogeneous distribution (i.e., across multiple domains or multiple classes). In this case, OCC methods have difficulty in reflecting the categorical information of the domain properly. To tackle this issue, we propose a two-stage framework that leverages the latent categorical information to improve representation learning for textual OOD detection. In the first stage, we train a transformer-based sentence encoder for *pseudo labeling* by contrastive loss and cluster loss. The second stage is *pseudo label learning* in which the model is re-trained with pseudo-labels obtained in the first stage. The empirical results on the three datasets show that our two-stage framework significantly outperforms baseline models in more challenging scenarios.

1 Introduction

Deep neural networks show outstanding performance on benchmark datasets that have the same training and test domains. However, once the model is deployed to the real world, it can face out-of-domain (OOD) instances that make the model predict unreliable outcomes related to AI safety issues (Amodei et al., 2016; Hendrycks and Gimpel, 2017). For this reason, the OOD detection task aims to discriminate whether given instances are from in-domain (IND) or not. One of the main OOD detection approaches is to use a classifier that predicts the labels of IND samples, based on the fact that the classifier has lower confidence in predicting the OOD samples than the IND (Hendrycks and Gimpel, 2017; Lee et al., 2018).

As this approach targets only supervised tasks that require IND labels to train the classifier, it has a limitation on unsupervised tasks. To overcome this problem, recent studies have proposed unsupervised OOD detection (or the without label scenario) that can be utilized in a more general use case (Xu et al., 2021; Jin et al., 2022). This setting can be regarded as one-class classification (OCC) because it uses only IND instances without labels and aims to distinguish novel samples from IND instances. Within this background, unsupervised OOD detection methods introduce OCC approaches such as OC-SVM and SVDD (Xu et al., 2021; Sohn et al., 2020). Meanwhile, self-supervision based models exploit a novel property named inlier priority (Wang et al., 2019) by using pseudo labels that are generated for surrogate supervision (e.g., rotation transformation (Hendrycks et al., 2019)).

In the field of natural language processing (NLP), this approach is adopted in combination with self-supervised methods of pretrained-language models (Manolache et al., 2021a). However, there are tasks where the categorical labels of training data are not available, while the IND has categorical distributions (e.g., summarization, topic modeling). OCC methods can suffer in this scenario (Jin et al., 2022; Park et al., 2021) due to the absence of IND labels, because it is difficult for the model to explicitly reflect the latent categorical distribution.

To tackle this problem, we propose a two-stage framework for textual out-of-domain detection that embeds similar INDs close together by considering latent categorical information of heterogenous IND instances without labels, and then detects OOD instances based on the learned embedding space. To achieve this, in the first stage, we conduct *pseudo labeling* of training samples by using an unsupervised clustering method combined with contrastive

* indicates corresponding author.

Findings of the Association for Computational Linguistics: EACL 2023, pages 1031–1041
May 2-6, 2023

loss. Next, the model from the first stage is refined by the given pseudo labels, which we call *pseudo label learning* (PLL). We find that this second stage of PLL greatly improves the representation learning for IND instances. After training is done, the inference step uses a confidence score function that measures the likelihood of whether an input is IND or OOD.

Our experimental results on three real-world datasets with the pre-trained RoBERTa (Liu et al., 2019) as a base architecture show that the proposed framework substantially outperforms the baseline models in various settings. In addition, we conduct embedding space analysis to confirm the effectiveness of PLL and show that it learns a more suitable representation for OOD detection by increasing inter-cluster distance significantly, which makes OOD samples more distinct from the clusters.

In summary, our main contributions are as follows:

- We propose a new framework for text OOD detection that effectively utilizes latent categorical information of IND through two successive steps of clustering for obtaining pseudo labels and then re-learning the pseudo labels for better representation learning.
- We provide a systematic analysis of the result by dividing OOD instances into near-OOD and far OOD depending on how close they are to IND samples. Our method works especially well on near-OOD, a more challenging scenario, in comparison with other methods. We also analyze the embedding space to confirm the effectiveness of our PLL approach.
- We empirically demonstrate that our proposed method is highly effective in multi-domain settings where the IND distribution has high variability, by increasing the inter-cluster distances and placing OOD out of detection boundaries of each cluster.

2 Related Work

Out of Domain Detection. OOD detection aims to distinguish OOD instances from IND to prevent a model trained for IND from making wrong predictions in the real applications. One of the main approaches is to rely on a classifier for IND labels, supposing that the softmax probability value of the IND will be larger than OOD (Hendrycks and Gimpel, 2017). Furthermore, (Liang et al., 2018; Lee et al., 2018; Hsu et al., 2020) improve this method by adding perturbation to the inputs, which further increases the softmax probability of IND. In the NLP field, Hendrycks et al., 2020 find out that transformer-based models are more effective than convolutional neural networks (LeCun et al., 1998) or long short-term memory (Hochreiter and Schmidhuber, 1997) based models in detecting textual OODs. To improve OOD detection performance for the models, (Zhou et al., 2021) utilize supervised contrastive loss that creates a more compact representation. However, these approaches cannot be used without IND labels.

Unsupervised Out of Domain Detection. Self-supervised methods can handle this issue by using augmentation techniques (Sehwag et al., 2020; Wang et al., 2019). Manolache et al., 2021a adopt this approach by utilizing the training scheme introduced in ELECTRA (Clark et al., 2019). They use a generator to replace random masked tokens in the input and train a discriminator to predict whether each token is replaced by the generator or not. Xu et al., 2021 focus on the findings that different layers of BERT Devlin et al., 2019 can capture different linguistic information. They compute the Mahalonobis distance using the embeddings in each layer and construct a new vector consisting of the distance values across all the layers. This new feature vector is used as input to OCC-based OOD detection methods. However, these models are difficult to perform well when INDs are in heterogeneous domains (or multiple classes), because they do not explicitly reflect the multimodal IND distribution. Cluster-based approaches can help alleviate this problem since they assume that the IND has a latent class distribution in its feature space. Jin et al., 2022 introduce a clustering method for representation learning to reflect categorical distributions on the embedding space. Our approach is motivated by (Jin et al., 2022), but our method generates pseudo labels and uses them explicitly to reinforce this categorical information, which greatly improves the performance.

3 Proposed Framework

In this section, we describe our two-stage framework for unsupervised OOD detection. First, the purpose of stage 1 is to generate pseudo labels that include categorical information of IND samples. We train a sentence encoder based on a pre-trained

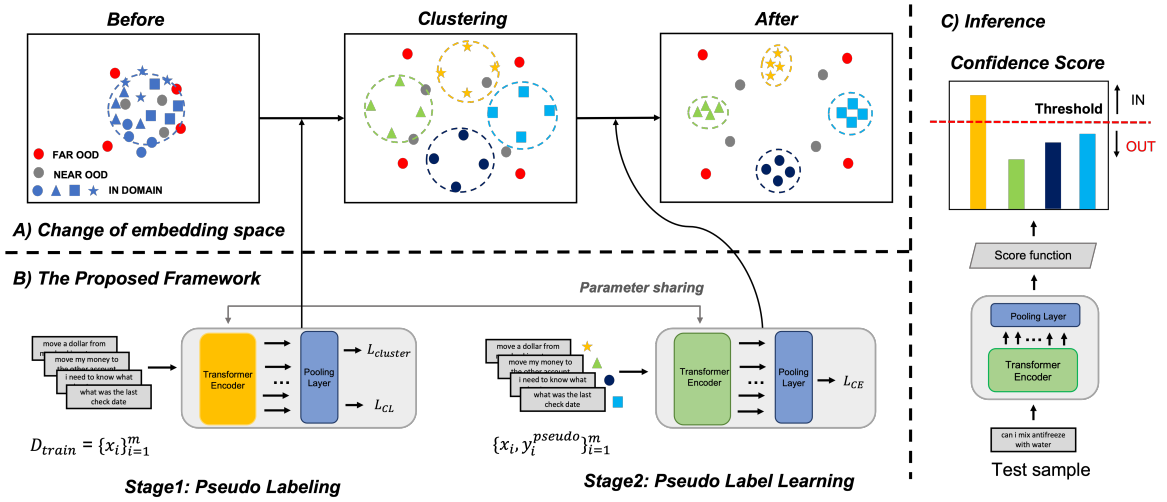

Figure 1: The overall framework of the proposed method. A) illustrates the change of representations in the embedding space during two stages of the training phase. B) shows the proposed framework. It consists of two stages of *pseudo labeling* (stage1) and *pseudo label learning* (stage2). C) shows the inference phase to detect OOD samples using a confidence score function. D_{train} is a training dataset that contains only IND samples x_i, without labels. y_i^{pseudo} is the pseudo label for x_i generated in stage 1.

transformer for *pseudo labeling* of IND training samples using contrastive loss and cluster loss. After then, in stage 2, we perform *pseudo label learning*, designed to explicitly utilize the pseudo labels for reinforcing the categorical information through a classification task. Finally, we use a scoring function that indicates the confidence of being IND to detect OOD samples at test time. Our proposed framework is illustrated in Figure 1.

3.1 Pseudo Labeling

The *pseudo labeling* stage is designed to generate pseudo labels y_i^{pseudo} for each x_i in $D_{train} = \{x_i\}_{i=1}^{M}$. To do that, we assume that IND data have K categories that are represented in the latent semantic space. Let μ_k denote the centroid of each cluster k and ψ be a transformer-based sentence encoder:

$$e_i = \psi(x_i).$$

For each sample x_i, we use the Student's t-distribution to compute a soft assignment probability q_{ik} , meaning the probability that the sample i belongs to the cluster k, by the following equation (Van der Maaten and Hinton, 2008):

$$q_{ik} = \frac{(1 + ||e_i - \mu_k||_2^2])^{-\frac{\alpha+1}{2}}}{\sum_{\acute{k}=1}^{K}(1 + ||e_i - \mu_{\acute{k}}||_2^2])^{-\frac{\alpha+1}{2}}},$$

Here, α represents the degrees of freedom of the Student's t-distribution. In this work, we set $\alpha = 1$. The cluster centroids and the soft assignment probability can be refined iteratively by using an auxiliary target distribution proposed by (Xie et al., 2016) as:

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

where $f_k = \sum_{j=1}^{M} q_{jk}$ is the soft cluster frequency to normalize q_{ik} raised to the second power. This target distribution first sharpens the soft assignment probability q_{jk} by raising it to the second power and then normalizes it by the associated cluster frequency. The soft assignment is optimized based on KL-divergence between $p_i = (p_{i1}, ..., p_{iK})$ and $q_i = (q_{i1}, ..., q_{iK})$:

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

The clustering objective is then defined as follows:

$$L_{cluster} = \frac{1}{M} \sum_{i=1}^{M} l_i^C$$

This loss function encourages learning from cluster assignment with high confidence and debiasing imbalanced cluster assignment.

Following (Zhang et al., 2021), we also adopt contrastive learning to improve clustering performance. Contrastive loss scatters the samples while closely embedding samples sharing the same properties. For contrastive learning, we use dropout

mask augmentation which simply feeds the same input to the transformer-based encoder[1] twice(Gao et al., 2021). Using this augmentation method, we construct a positive pair (x_i^0, x_i^1) from the same x_i with different dropout masks. We try to minimize the following contrastive learning loss:

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

where $z_i = g(\psi(x_i))$ and g is a network of fully-connected layers. We choose sim$(\cdot)$ as the dot product between a pair of normalized outputs, i.e., $sim(z_i, z_j) = z_i^T z_j / ||z_i||_2 ||z_j||_2$. Then the overall contrastive learning objective is defined as:

$$L_{CL} = \frac{1}{2M} \sum_{i=1}^{2M} l_i^{CL}$$

In summary, the final objective for stage 1 is the following:

$$L_{stage1} = L_{cluster} + \lambda L_{CL} \quad (1)$$

After training the model for pseudo labeling by using the stage1 loss, we assign the pseudo label y_i^{pseudo} for each $x_i \in D_{train}$ using the soft assignment probability.

3.2 Pseudo Label Learning

Contrastive learning is useful for clustering and pseudo labeling because contrastive loss separates samples apart from each other to prevent overlap in the representation space. However, it is not sufficient for OOD detection because OOD samples can be located close to the cluster boundaries as illustrated in Figure 1. Therefore, we introduce *pseudo label learning*(PLL), which allows the text encoder to learn representations that are more suitable for OOD detection. PLL explicitly uses pseudo labels to further separate clusters in the embedding space. Therefore, we fine-tune the model by targeting pseudo labels y_i^{pseudo} using the cross-entropy loss. The loss function in stage 2 is as follows:

$$L_{stage2} = L_{CE} = -\sum_{i=1}^{M} y_i^{pseudo} \cdot \log(p_i)$$

where p_i is the predicted probability distribution for the pseudo label.

[1]Transformer already has dropout mask in fully-connected layer and attention probabilities

3.3 Confidence score function

Next, we introduce the confidence score function s for OOD detection that uses a classifier in stage 2. The scoring function s aims to map the representations of instances to confidence scores, where higher scores indicate higher confidence for being IND. In the following, we present several options for this scoring function.

Maximum Softmax Probability (MSP). Hendrycks and Gimpel, 2017 suggest the maximum class probability among K training classes in the softmax layer as an OOD indicator. This method has been extensively used as a baseline for OOD detection (Hendrycks et al., 2020; Zhou et al., 2021), which defines the score as:

$$s = 1 - \max\{p_k \mid k = 1, ..., K\}.$$

Energy Score (Energy). Liu et al., 2020 propose energy based score that theoretically outperforms the softmax based score, which is defined as:

$$s = -\log \sum_{j}^{K} (w_j^T h)$$

where w_j is the weight of the j^{th} class in the softmax layer, and h is the input to the softmax layer. A higher s means higher probability density in OOD classes and thus implies lower IND likelihood.

Mahalanobis Distance (Maha). Podolskiy et al., 2021 showed that the distance-based scoring function can outperform other methods in a supervised setting, which is defined as:

$$s = -\min_k (h - \mu_k)^T \Sigma_k^{-1} (h - \mu_k)$$

where μ_k is the mean vector and $\sum_k$ is the covariance matrix of each latent class k. Then, given an instance x during inference, it calculates the confidence score as the minimum Mahalanobis distance among the K classes.

4 Experiments

In this section, we present the experimental setting for the evaluation of the proposed framework. We describe the used datasets and how to construct IND and OOD samples under unsupervised OOD scenarios.

4.1 Dataset

To evaluate the proposed model, we select the following three real world datasets.

Dataset	Ratio	Model	AUROC	AUIN	AUOUT	AUROC	AUIN	AUOUT
			Near OOD			Far OOD		
CLINC150	**0.25**	**DATE**	74.30	49.23	88.43	88.03	89.55	84.68
		MDF	79.51	59.43	91.60	91.19	90.54	88.61
		Ours	**93.68**	**86.73**	**97.39**	**98.46**	**98.7**	**98.11**
	0.5	**DATE**	69.67	67.53	68.23	86.71	93.09	72.37
		MDF	73.81	70.51	72.85	87.81	93.77	75.27
		Ours	**89.72**	**89.71**	**88.9**	**97.00**	**98.56**	**94.04**
	0.75	**DATE**	66.88	83.22	41.34	86.38	95.03	63.83
		MDF	69.42	85.78	45.83	83.18	93.83	59.05
		Ours	**87.21**	**94.51**	**71.69**	**96.52**	**98.84**	**91.1**
HWU64	**0.25**	**DATE**	69.85	44.43	85.77	79.36	59.23	91.00
		MDF	77.19	60.51	88.95	85.15	73.5	93.62
		Ours	**85.25**	**72.25**	**92.84**	**91.69**	**82.97**	**96.69**
	0.5	**DATE**	64.82	64.04	61.68	79.78	72.68	83.96
		MDF	68.60	70.22	65.37	82.32	77.22	86.55
		Ours	**81.43**	**81.99**	**79.12**	**91.39**	**87.08**	**94.15**
	0.75	**DATE**	63.03	82.97	34.58	81.55	80.19	79.74
		MDF	66.96	85.84	36.41	83.986	83.77	83.03
		Ours	**78.63**	**91.11**	**52.88**	**90.46**	**89.33**	**90.76**
BANKING77	**0.25**	**DATE**	75.36	44.24	90.34	98.41	97.7	98.9
		MDF	70.81	55.62	70.51	99.42	99.09	99.56
		Ours	**88.72**	**77.04**	**95.31**	**99.83**	**99.72**	**99.82**
	0.5	**DATE**	66.70	61.76	68.84	98.21	98.56	97.78
		MDF	64.73	63.46	63.48	99.14	99.11	98.73
		Ours	**78.63**	**77.34**	**79.3**	**99.21**	**99.38**	**98.95**
	0.75	**DATE**	60.65	79.25	38.31	97.94	98.83	96.64
		MDF	61.61	81.43	35.66	**98.94**	98.71	**98.45**
		Ours	**70.34**	**85.87**	**46.51**	98.57	**99.27**	97.46

Table 1: OOD detection performance with different IND class ratios (25%, 50%, and 75%) on three datasets, CLINC150, HWU64, and BANKING77. Scores in bold type are the best results. For all of our methods, we report the averaged results using Mahalanobis distance-based score and the number of clusters equal to the number of IND classes due to space limitations. We collected the results for other methods (Xu et al., 2021, Manolache et al., 2021b) by running their released codes.

CLINC150 (Larson et al., 2019) is a dataset designed for OOD detection. The training set contains 15,000 utterances with 10 domains and 150 classes (e.g., travel.timezone, home.reminder, and credit_cards.rewards_balance). This dataset also provides 1,000 OOD samples that are not within any of 150 classes. For evaluation, we use 4,500 IND and 1,000 OOD samples from the test set.

HWU64 (Xingkun Liu and Rieser, 2019) includes 8,954 utterances for 64 intents with 21 domains (e.g., alarm_set, cooking_recipe, and calendar_query). For evaluation, we use 1,076 IND samples from the test set.

BANKING77 (Casanueva et al., 2020) contains 8,622 utterances related to banking with 77 different fine-grained intents in the training set. Despite consisting of a single domain, this dataset is challenging, as it requires fine-grained differentiation between very similar intents. For evaluation, we use 3,080 IND samples from the test set.

4.2 Experimental setting

We carefully design experimental scenarios assuming that training data consist of instances distributed across multiple domains with any category given. Inspired by Zhang et al., 2022, we divide OOD samples into two types: near-OOD and far-OOD. We suppose that the near-OOD samples are distributed in the same domain with the training samples but labeled as different categories, whereas the far-OOD samples are distributed in distinct domains. The proposed scenarios are more challenging because OOD can share characteristics with IND.

For our scenarios, we randomly select a subset of classes in the training data as IND, with IND class ratios of 25%, 50%, and 75% and use the remaining classes as near-OOD. Following (Zhang et al., 2022), we use the OOD samples in the CLINC150 dataset as far-OOD. We split each dataset five times with different random seeds, which are shared

across all the models for a fair comparison.

4.3 Baselines

We compare our method with the following unsupervised OOD detection methods: MDF (Xu et al., 2021) and DATE (Manolache et al., 2021a). **MDF** utilizes full features from all the layers of a pretrain-transformer model and calculates the Mahalanobis distance vector from the layer representations, which is in turn used as input to OC-SVM. In addition, there are additional training stages such as IMLM (In-domain Masked Language Model) and BCAD (Binary Classification with Auxiliary Dataset) before feature extraction. **DATE** is a pseudo label based approach. It uses a self-supervised learning method of ELECTRA that distinguishes whether each token is replaced or not to generate anomaly scores from the loss obtained by pseudo-labeled tokens.

4.4 Evaluation Metric

To evaluate our proposed method, we report three different metrics following (Liang et al., 2018; Xu et al., 2021). The area under the receiver operating characteristic curve (AUROC) depicts the relationship between the true positive rate and the false positive rate. A higher score indicates improved distinguishment between IND and OOD by the model. The area under the precision-recall curve (AUPR) shows the precision and recall against each other, for IND and OOD testing sentences, denoted by AUIN and AUOUT, respectively.

4.5 Implement details

For a fair comparison, we also select *roberta-base* from Huggingface's Transformers (Wolf et al., 2020) as a base architecture for the sentence encoder, the same as MDF. In stage 1, we choose $\tau = 0.5$, $\lambda = 10$, and $\alpha = 1$. We use a constant learning rate of 3e-6 to optimize the sentence encoder and 3e-4 to optimize $g()$ and the liner layer for soft cluster assignment. In stage 2, we set the learning rate to 3e-5. We use the Adam optimizer (Kingma and Ba, 2015) with a batch size of 128 for both stages. We used the same hyperparameters for all datasets and splits following Manolache et al., 2021a.

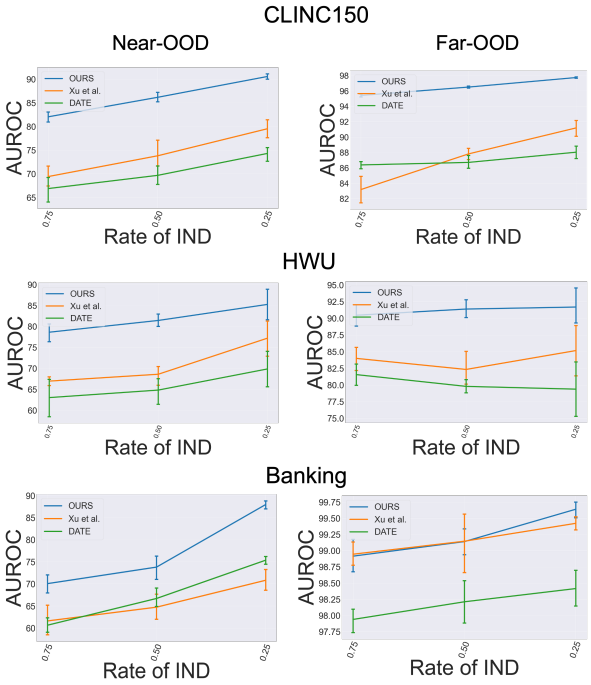

Figure 2: The OOD detection performance with respect to different ratios of IND classes (0.25%, 0.50%, and 0.75%).

5 Result

5.1 Comparisons with baseline methods

Table 1 presents the performance of each method on the three datasets with different IND class ratios (25%, 50%, and 75%). The proposed framework outperforms two baselines, DATE and MDF, by a large margin for the AUROC, AUIN, and AUOUT scores across all three datasets regardless of IND class ratios in the near-OOD and far-OOD setting, except just one case (BANKING77 with the ratio 0.75). In particular, our method greatly improves the performance over other methods on the near-OOD dataset, which represents a more challenging scenario. This shows that the proposed method is robust in multi-domain IND settings regardless of OOD types. In HWU64 dataset that contains more heterogenous domains than the other two datasets, the OCC-based models, MDF and DATE, appear to have weaknesses in more heterogeneous domain settings, but our method shows good performance. In addition, in the BANKING77 that is the least heterogeneous setting, our method shows similar or higher performance than the other methods as well.

Figure 2 shows the performance with respect to the IND class ratio on three datasets. The performance of all models tend to increase when the ratio

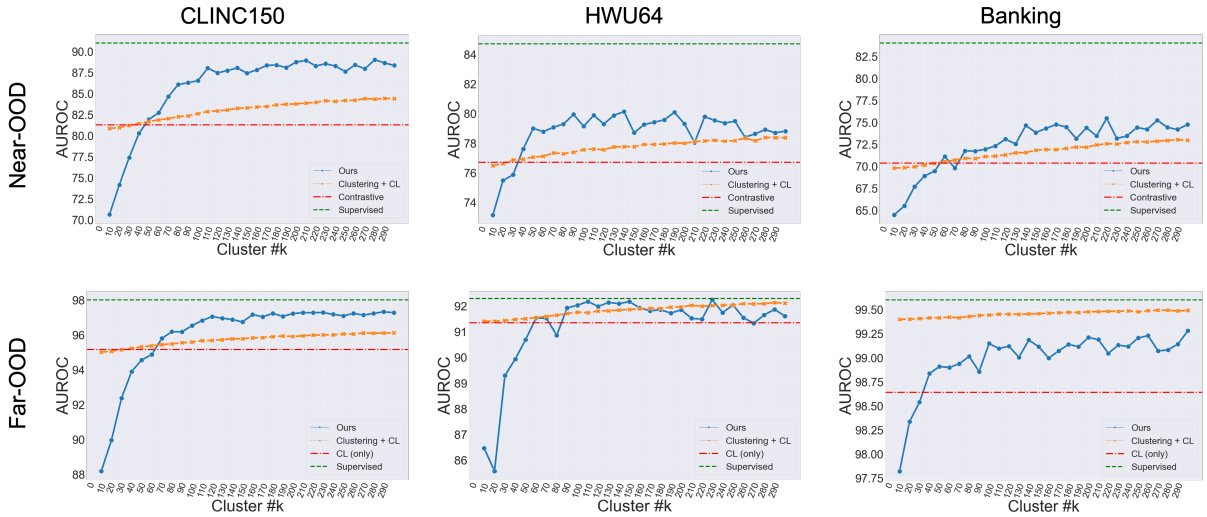

Figure 3: Effect of the number of clusters K with an ablation study. The first row is for the near-OOD setting, the second row is for the far-OOD setting. The columns sequentially correspond to the results on CLINC150, HWU64, and Banking77 datasets for the IND class ratio of 0.75. We compare our proposed method (shown in blue) with the result using only L_{stage1} (orange) and the result using only L_{CL} (red). The green line denoted as *supervised* shows the result when the ground truth labels of IND classes are used during training.

decreases, which is as expected, because fewer IND classes imply less heterogeneous IND distributions and hence easier scenarios. In addition, our method shows more accurate and robust performance with smaller variances (shown as vertical line segments). However, OCC-based methods are more susceptible to randomness during training because they need to bind one characteristic.

5.2 Number of clusters K

The selection of the number of clusters K is an open problem for unsupervised OOD detection since there is no validation OOD set to choose the hyperparameter value. For the results shown in Table 1 and Figure 2, we set K as the number of IND labels given. To measure the influence of K, we plot the change of performance as K is increased in Figure 3. The blue curve indicates our method and the orange line indicates clustering based method with clustering loss and contrastive loss. We find that a larger number of clusters K generally leads to better results for OOD detection. As K increases, the blue curve moves upward to the right, showing that the larger number of clusters allows more detailed consideration of IND samples. It allows more OOD samples to be pushed away from the clusters. In other words, OOD samples that are placed inside a cluster can be located in between as the clusters become more segregated. Therefore, choosing an appropriately large K is advantageous for OOD detection. This is empirically demonstrated in Figure 3.

Dataset	Model	Near OOD	Far OOD
CLINC150	**MSP**	76.51	89
	Energy	78.55	91.53
	Mahalanobis	**87.21**	**96.52**
HWE	**MSP**	69.32	78.34
	Energy	71.32	83.31
	Mahalanobis	**78.63**	**90.46**
BANKING77	**MSP**	58.62	88.07
	Energy	58.87	92.23
	Mahalanobis	**70.34**	**98.57**

Table 2: Performance comparison using different confidence score functions. In this result, we set the number of clusters K equal to the number of IND classes in each dataset

5.3 Ablation study

As shown in Figure 3, our two-stage approach combining clustering and PLL outperforms clustering-based approaches (shown in orange and red) especially on near-OOD setups. This result reveals that PLL at the second stage utilizes more categorical information than the clustering-based models in stage 1. In far-OOD, our method shows lower performance in only one case (BANKING77) with a very small margin (less than 0.5%). The green line indicates the performance of the oracle model that is supervised by ground truth labels of IND samples during training. In CLINC150, the perfor-

Dataset	Model	max	min	mean	median
CLINC150	Clustering(Only)	4.268	2.464	3.508	3.566
	PLL	18.776	4.46	9.92	8.968
HWU	Clustering(Only)	7.523	3.577	5.439	5.35
	PLL	16.997	5.588	10.082	9.153
BANKING	Clustering(Only)	5.054	3.706	4.428	4.386
	PLL	16.44	5.654	12.1	12.114

Table 3: Intra-cluster variance statistics

mance of our proposing model with high enough k can be almost close to the green line in the near-OOD setting. In addition, our methods show similar performance with the supervised model on the HWU64 dataset in far-OOD settings.

Regarding to the choice for a scoring function, Mahalanobis distance shows the best result regardless of datasets and OOD settings (Table 2). This is because MSP and energy-based methods are based on the predicted class probabilities while pseudo labels can contain errors. In contrast, Mahalanobis distance is based on representations, so it can be more robust to clustering results even when there are miss-labeled instances.

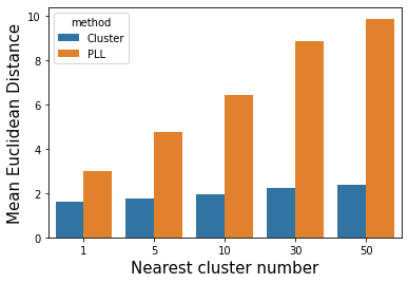

Figure 4: Inter-cluster distance statistics with different numbers of nearest cluster centers.

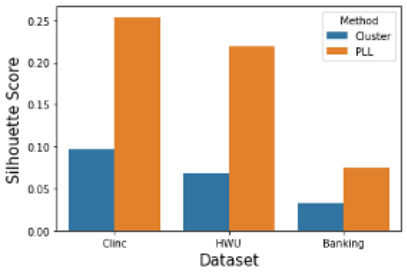

Figure 5: Average silhouette score before and after PLL.

5.4 Analysis of representation space

To investigate why our PLL approach improves OOD detection performance over clustering-based methods, we additionally examine three metrics: intra-cluster variance, inter-cluster distance, and

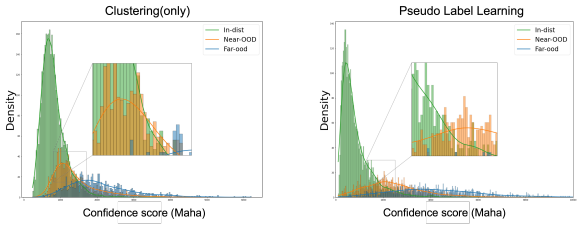

Figure 6: Distributions of the confidence scores before PLL (left) and after PLL (right) on CLINC150 dataset with IND class ratio of 0.75. The confidence score distribution is shown in green for IND, yellow for near-OOD, and blue for far-OOD.

the silhouette score. Table 3 shows the statistics of intra-cluster variance, which can indicate the degree of clustering of the data representations within a cluster. Specifically, we average the distances of the representations of samples with the same pseudo label to the cluster center in the test set as intra-cluster variance, then report min/max/mean/median values on all clusters. And Figure 4 shows the inter-class distances. We average dot product distances between each class center to its C nearest class centers, then average results from all classes as inter-class distance. The x-axis denotes the number of nearest centers C. We find that the intra-cluster variance becomes higher when the clustering is followed by PLL, which means PLL can ruin intra-cluster distribution. However, the inter-cluster distances are also significantly increased through PLL. To find out the balance of the two distances, we compare the silhouette scores before and after PLL in Figure 5, which shows that PLL improves the silhouette scores by a large margin. This implies that PLL can make the clusters far apart from each other and therefore OOD sample to be placed in between the clusters.

5.5 Visualizations

We visualize the confidence score distributions to confirm the effectiveness of our PLL scheme. Figure 6 shows the confidence score distribution on CLINC150 test set with the IND ratio of 0.75. Although the score distributions of near-OOD and IND still overlap when we apply clustering only, after performing PLL, the score distribution for IND shifted to the left, while the distributions of both OOD samples shifted to the other side. Therefore, the score distributions become more discriminable between IND and OOD samples through PLL.

6 Conclusion

In this work, we proposed a two-stage framework for unsupervised OOD detection that effectively utilizes the categorical information of IND instances by pseudo labeling and pseudo label learning. In addition, for a more systematic analysis of OOD performance, we introduced the near-OOD setting, which is a more challenging yet realistic scenario. In most of our experimental settings, our framework outperforms the baseline models with significant margins. We further justify the improvement of the proposed model's OOD detection performance by analyzing the embedding space with inter or intra-cluster distances and silhouette scores. In future work, we will further investigate how to reduce intra-cluster variations while maintaining inter-cluster distances.

Limitations

The proposed methods show relatively stable performance with respect to the number of clusters (K), but it still has a limitation of choosing the optimal one. In particular, we conduct the experiment by setting the maximum value of K to 300. However, a too large K can degrade the model performance by reducing the number of samples per cluster for classification in stage 2. In addition, since the proposed framework depends on a clustering method, its performance can be limited by the clustering performance. Experiments are only conducted on three intent task datasets due to the near-OOD and the far-OOD settings in heterogeneous domains. We remain those limitations for future works.

Acknowledgements

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT)(No. NRF-2022R1A2C1007434), and also by the Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korea Government (MSIT) (Artificial Intelligence Innovation Hub) under Grant 2021-0-02068.

References

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

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

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.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

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

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.

Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. 2019. Using self-supervised learning can improve model robustness and uncertainty. *Advances in neural information processing systems*, 32.

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

Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. 2020. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10951–10960.

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

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

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31.

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

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

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Andrei Manolache, Florin Brad, and Elena Burceanu. 2021a. 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.

Andrei Manolache, Florin Brad, and Elena Burceanu. 2021b. 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.

Junekyu Park, Jeong-Hyeon Moon, Namhyuk Ahn, and Kyung-Ah Sohn. 2021. What is wrong with one-class anomaly detection?

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.

Vikash Sehwag, Mung Chiang, and Prateek Mittal. 2020. Ssd: A unified framework for self-supervised outlier detection. In *International Conference on Learning Representations*.

Kihyuk Sohn, Chun-Liang Li, Jinsung Yoon, Minho Jin, and Tomas Pfister. 2020. Learning and evaluating representations for deep one-class classification. In *International Conference on Learning Representations*.

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

Siqi Wang, Yijie Zeng, Xinwang Liu, En Zhu, Jianping Yin, Chuanfu Xu, and Marius Kloft. 2019. Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network. *Advances in neural information processing systems*, 32.

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

Junyuan Xie, Ross Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pages 478–487. PMLR.

Pawel Swietojanski Xingkun Liu, Arash Eshghi and Verena Rieser. 2019. Benchmarking natural language understanding services for building conversational agents. In *Proceedings of the Tenth International Workshop on Spoken Dialogue Systems Technology (IWSDS)*, pages xxx–xxx, Ortigia, Siracusa (SR), Italy. Springer.

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.

Dejiao Zhang, Feng Nan, Xiaokai Wei, Shang-Wen Li, Henghui Zhu, Kathleen Mckeown, Ramesh Nallapati, Andrew O Arnold, and Bing Xiang. 2021. Supporting clustering with contrastive learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5419–5430.

Jianguo Zhang, Kazuma Hashimoto, Yao Wan, Zhiwei Liu, Ye Liu, Caiming Xiong, and S Yu Philip. 2022. Are pre-trained transformers robust in intent classification? a missing ingredient in evaluation of out-of-scope intent detection. In *Proceedings of the*

4th Workshop on NLP for Conversational AI, pages 12–20.

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.