

Enriching Category Representations with LLMs Towards Robust Zero-Shot Out-of-Distribution Detection

Dian Chao¹, Yuxuan Zhang¹, Luping Zhou², and Yang Yang¹ (✉)

¹ Nanjing University of Science and Technology, Nanjing 210000, China
{chaodian,xuan_yuzhang,yyang}@njjust.edu.cn

² University of Sydney, Sydney 2006, Australia luping.zhou@sydney.edu.cn

Abstract. Recent advancements in foundation models, particularly Visual-Language Models (VLMs) have enabled effective zero-shot Out-of-distribution (OOD) detection. Existing methods attempt to generate the names of OOD classes similar to ID classes to explore the textual space of VLMs. However, they fail to integrate relevant in-distribution (ID) information to reveal specific OOD features, thus limiting the distinction between ID and OOD classes. To address this issue, we propose a novel zero-shot OOD detection approach by leveraging LLMs to capture nuanced textual features of both ID classes and their OOD counterparts, enabling VLMs to focus on specific regions of images and improve zero-shot OOD detection performance. Specifically, we prompt LLMs with extensive world knowledge to generate descriptive terms for ID classes. Moreover, LLMs are also employed to generate the names of challenging OOD classes and detailed descriptions that distinguish OOD from ID classes, significantly improving the separation between ID and OOD classes. Subsequently, a score is designed by adjusting the confidence of the ID classes to detect OOD samples. Experiments demonstrate that our method achieves state-of-the-art (SOTA) performance on multiple OOD detection benchmarks. The code is available at <https://anonymous.4open.science/r/zs-clip-ood-859E>.

Keywords: Out-of-distribution Detection · Zero-shot Learning · Multi-modal Data.

1 Introduction

Machine learning models trained only on ID data often perform well in controlled settings. However, these models frequently misclassify OOD data as ID in open-world environments [1–6]. In classification tasks, failing to recognize OOD samples can lead to severe consequences, particularly in critical fields such as autonomous driving [7, 8] and medical diagnostics [9]. Therefore, detecting and excluding OOD data is essential for maintaining the reliability and safety of these systems.

Previous works focus on analyzing the uncertainty of the model’s predictions to determine whether a sample belongs to an unknown class [10–14]. In the

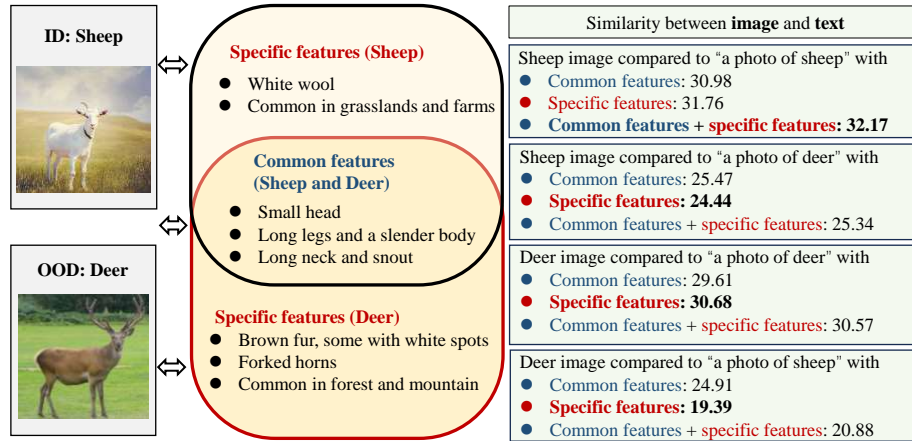


Fig. 1. Similarity between Images of Sheep/Deer and Descriptive Texts: Comparison of Common and Specific Features. ‘Specific features’ refer to the unique characteristics of sheep or deer, while ‘common features’ represent the shared traits. The left side shows the similarity scores between images of sheep/deer and text descriptions that include ‘common’, ‘specific’, or ‘common+specific’ features. For each image, we aim to maximize similarity with the corresponding class description while minimizing similarity with descriptions of unrelated classes.

field of image analysis, early approaches rely on the model’s ability to extract image features but lack the utilization of textual information associated with the categories [15, 16]. With the development of foundation models, VLMs [17] exhibit strong generalization capabilities after being trained on large-scale image-text pairs. In recent years, an increasing number of works [18, 19] have focused on leveraging textual modality features for OOD detection using VLMs. These approaches demonstrate superior performance compared to previous OOD detection methods. However, these methods primarily utilize ID class names and lack comprehensive exploitation of the textual modality. Recent works have begun to explore more extensive information from the textual modality. Several approaches [15, 16] endeavor to generate OOD class names using resources such as WordNet [20], whereas others [21] leverage LLMs [22] to produce semantic descriptions for ID classes.

In practical scenarios, this approach often fails with hard OOD samples due to their high similarity to ID samples. Leveraging VLMs’ ability to align textual and visual features, we prompt LLMs to generate OOD descriptions and measure their similarity to images. Possibly, here give an example of a common feature and an example of a specific feature by combining descriptive terms (e.g., “a photo of a sheep with white wool, commonly found in grasslands and farms”) and computing similarity with both sheep and deer images, we observe that common features lead to misclassification. In contrast, more specific features help reduce it. Acquiring such fine-grained textual descriptions of OOD classes can

significantly enhance OOD detection performance. Unfortunately, large lexical databases like WordNet, while useful for constructing categorical relationships, lack the contextual specificity needed to capture specific features for each category. With the advancement of LLMs trained on extensive text datasets, these models have acquired broad knowledge and the capability to analyze relationships and distinctions between categories. In this paper, we harness the power of LLMs to generate fine-grained textual descriptions.

This work introduces a method that utilizes LLMs to generate names for hard OOD classes resembling ID classes while systematically excluding synonyms and near-synonyms through similarity calculations. To enhance VLMs’ OOD detection capacities, fine-grained descriptions are generated by simultaneously considering both ID and OOD class names. Specifically, this work uses LLMs to generate descriptions for ID classes. Subsequently, LLMs are also employed to generate OOD classes that are prone to be misclassified as the given ID classes, along with specific features that distinguish these hard OOD classes from their ID counterparts. This approach aims to maximize the separation of textual feature spaces between hard OOD and ID classes. Furthermore, the impact of shared features between OOD and ID classes on OOD detection performance is analyzed using information entropy, demonstrating that common features adversely affect OOD detection. To address this, we propose a novel scoring method, adjusting the confidence of ID samples based on their similarity to hard OOD classes. Extensive experiments demonstrate that our method achieves SOTA performance across multiple datasets. This approach enhances the model’s performance in hard OOD detection tasks and exhibits strong generalization capabilities. In summary, our key contributions are as follows:

- We leverage LLMs to enrich the text features for ID samples. Specifically, we generate hard OOD class names for each ID class and identify specific features that distinguish OOD classes from their ID counterparts, maximizing the separation of the feature spaces between ID and OOD.
- The impact of common and specific features on OOD detection performance is analyzed using information entropy. In addition, we introduce a novel scoring method that adjusts the confidence scores of ID samples based on the similarity to generated OOD classes.
- The effectiveness of the proposed method is validated across diverse settings, including both easy and hard OOD tasks. Experimental results demonstrate that this approach achieves SOTA performance on multiple OOD detection benchmarks.

2 Related Work

Traditional OOD Detection is typically categorized into two types: training-time regularization [23–27] and post hoc methods [3, 13, 14, 28–30]. Training-time regularization methods assume that a subset of OOD data is accessible during model training. CSI [23] enhances the OOD detector through the application of contrastive learning. MOS [24] pre-groups all categories and introduces an

additional class to each group, redesigning the loss function for training. VOS [25] improves energy scores by generating virtual anomalies. LogitNorm [26] offers an alternative to cross-entropy loss by separating the influence of the logit norm from the training process. CIDER [27] improves OOD detection performance by optimizing contrastive loss.

Post hoc methods do not alter the model’s parameters; instead, they typically focus on designing an OOD score. MSP [3] utilizes the highest predicted softmax probability as the OOD score. ODIN [28] refines MSP by applying input perturbations and rescaling the logits. Energy [14] introduces the use of an energy function [31] to quantify OOD. Mahalanobis [13] calculates the OOD score based on the minimum Mahalanobis distance between the feature and the centroids of each class. GradNorm [29] develops the OOD score by utilizing the gradient space. ViM [32] integrates the norm of feature residuals with the principal space created by training features and the original logits to determine the degree of OOD-ness. KNN [30] explores the effectiveness of non-parametric nearest-neighbor distances for identifying OOD samples.

OOD Detection based on VLMs has been developed using CLIP [17] as the foundation, leveraging its powerful vision-language alignment capabilities. MCM [18] introduced this approach by utilizing maximum softmax probabilities to assess the similarity of images to known classes, thereby identifying OOD images. ZOC [33] train image decoders for extracting textual information from images. CLIPN [19] proposes constructing negative sample pairs and conducting pre-training to learn a ‘no’ concept for each class. Dai et al. [21] propose using LLMs to generate additional descriptive terms for ID classes to enrich textual semantic information. Recent studies [15, 16] have explored methods to leverage VLMs’ zero-shot inference capability by generating OOD categories through various approaches, aiming to represent potential OOD scenarios. Specifically, EOE [15] utilizes LLMs to generate potential outlier class labels and designs an outlier penalty function to detect OOD samples. NegLabel [16] acquisition utilizes WordNet to gather a diverse set of OOD category names, complemented by a scoring function to identify the OOD class with low similarity to current IDs.

Large Language Models such as GPT-3 [34], LLaMA-3 [35], GPT-4 [36], are leading advancements in natural language processing. These models are trained on massive datasets with parameters ranging from hundreds of billions to trillions. LLMs represent significant advancements in natural language processing, pushing boundaries in language understanding, generation, and adaptation across various domains. Given LLMs’ broad knowledge base, they are instrumental in providing similarities and differences among categories akin to ID.

3 METHODOLOGY

This section details the proposed approach. Section 3.1 defines the notation and outlines the problem. Section 3.2 introduces a method for generating outliers and fine-grained features by leveraging LLMs to augment class descriptions.

The complete framework is depicted in Figure 4. In Section 3.3, a novel OOD detection scoring function is presented, which clusters ID and OOD categories.

3.1 Notation and Preliminary

Without loss of generality, assume that we have n images which is denoted as $\mathbf{X} = \{x_1, \dots, x_n\}$. The ID class names set $\mathbf{Y}^{(\text{id})} = \{y_1^{(\text{id})}, \dots, y_c^{(\text{id})}\}$ is also available, where c denotes the number of class names. The goal of OOD detection is to determine whether an image $x \in \mathbf{X}$ belongs to the ID class $\mathbf{Y}^{(\text{id})}$ or not.

We prompt LLMs to generate OOD class names set to assist the OOD detection task. Additionally, extra information for both ID and OOD classes is generated to better align text and images. The descriptions for ID classes are denoted by $\mathbf{D}^{(\text{id})}$, and the OOD class names set and their descriptions are denoted by $\mathbf{Y}^{(\text{ood})}$ and $\mathbf{D}^{(\text{ood})}$, respectively. Furthermore, a pre-trained model is used to encode text including class name and description and image as feature, and then decide whether an image belongs to the ID class names set. Specifically, we use $\phi(\cdot)$ and $\psi(\cdot)$ to denote the image and text encoder, respectively. For an image \mathbf{x} and a text \mathbf{t} , their features can be calculated by:

$$\mathbf{u} = \phi(\mathbf{x}), \mathbf{v} = \psi(\mathbf{t}),$$

where d denotes the feature dimension, $\mathbf{u}, \mathbf{v} \in \mathcal{R}^d$ denote the image and text features, respectively. The text input can be a class name or description.

Based on the features of the given image and text information, we can design an evaluation function to decide whether the image belongs to the ID class or not.

3.2 Outliers and Fine-Grained Feature Generation

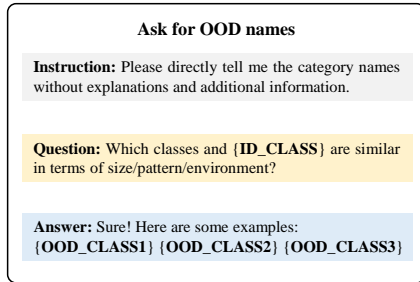


Fig. 2. The prompt queries to obtain OOD class names similar to ID classes, including instruction, question, and model response examples.

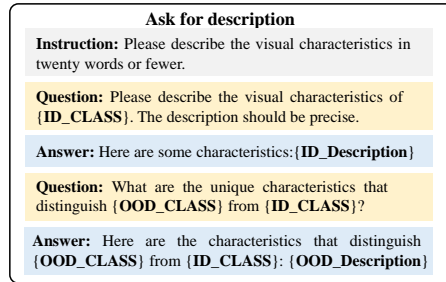


Fig. 3. The prompt queries to obtain descriptive words for ID and OOD classes, including instruction, question, and model response examples.

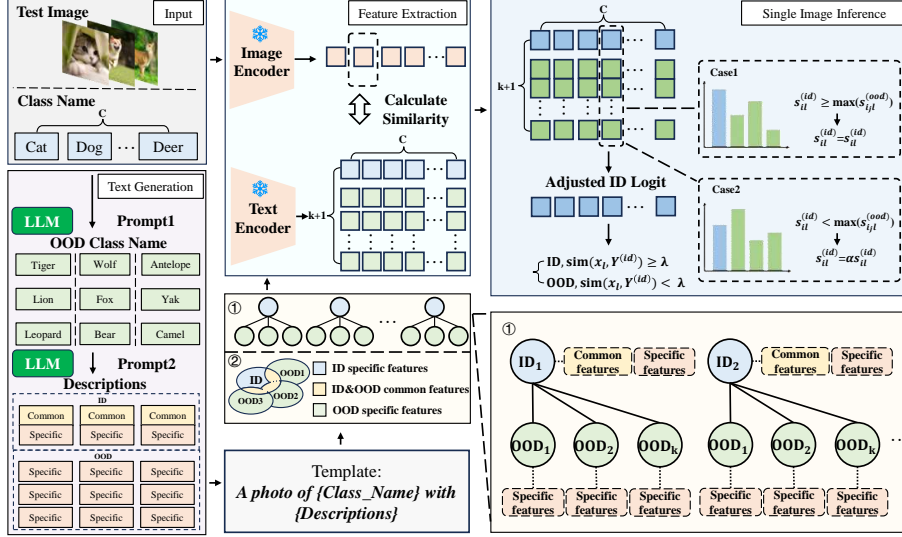


Fig. 4. The main framework of our model. LLMs are employed to generate OOD class names and descriptions for ID samples, following a fixed template. Feature vectors are then extracted using frozen encoders. Finally, the OOD detection process for a single image is demonstrated. In the enlarged section labeled ①, a detailed depiction is provided of how the relationship between an ID class and its corresponding hard OOD is constructed, along with their associated textual descriptions.

LLMs are adopted to generate additional textual information to support the OOD detection task. Beyond generating OOD class names, LLMs are also used to create nuanced textual descriptions for both ID and OOD class names.

LLMs are first utilized to generate OOD class names. To enhance the discriminative ability of the model, the generated OOD classes are required to closely resemble the ID classes. For example, if the ID class name is “cat”, an OOD class name like “tiger” is preferred over unrelated options such as “book”. To achieve this goal, prompts are refined to drive LLMs to generate target OOD class names by exploring the pivotal properties including size, pattern, and environment. The interaction process with LLMs is shown in Figure 2.

Formally, k OOD class names are generated for each ID class name, i.e., $\forall \mathbf{y}_i^{(id)} \in \mathbf{Y}^{(id)}$, we generate OOD class names set $\{\mathbf{y}_{i1}^{(ood)}, \dots, \mathbf{y}_{iK_i}^{(ood)}\}$. $\mathbf{Y}^{(ood)}$ are utilized to denote the whole OOD class names set, which is defined as follows:

$$\mathbf{Y}^{(ood)} = \bigcup_{i=1}^c \{\mathbf{y}_{i1}^{(ood)}, \dots, \mathbf{y}_{iK_i}^{(ood)}\},$$

where K_i denotes the number of generated OOD class names and its value is dependent on the output of LLMs.

Since the strong knowledge capacity of the LLMs, it is necessary to filter out some OOD classes [16] that are too similar to ensure the distinguishabil-

ity of the subsequent description generation. For a detailed explanation, please refer to Appendix A.1. To achieve this, features for both ID classes and their corresponding OOD class names are first extracted. Specifically, a pre-trained CLIP model is utilized to extract the textual features for a given ID class and its associated OOD class names:

$$\begin{aligned} \mathbf{v}_i^{(\text{id})} &= \psi(\mathbf{y}_i^{(\text{id})}), \\ \forall j \in \{1, \dots, k\}, \mathbf{v}_{ij}^{(\text{ood})} &= \psi(\mathbf{y}_{ij}^{(\text{ood})}). \end{aligned}$$

Then, the similarity between ID class and OOD class names is calculated

$$\forall j \in \{1, \dots, k\}, s_{ij} = \frac{[\mathbf{v}_i^{(\text{id})}]^\top \mathbf{v}_{ij}^{(\text{ood})}}{\|\mathbf{v}_i^{(\text{id})}\| \|\mathbf{v}_{ij}^{(\text{ood})}\|}.$$

According to similarity, for each ID class, the top- k OOD class names with the lowest similarity scores are selected to construct pairs, where $k \leq \min\{K_i\}_{i=1}^c$. This process results in the filtered OOD class name set:

$$\hat{\mathbf{Y}}^{(\text{ood})} = \bigcup_{i=1}^c \{\mathbf{y}_{il_1}^{(\text{ood})}, \dots, \mathbf{y}_{il_k}^{(\text{ood})}\},$$

where $|\hat{\mathbf{Y}}^{(\text{ood})}| = ck$.

After obtaining the ID class name $\mathbf{Y}^{(\text{id})}$ and the filtered OOD class name $\hat{\mathbf{Y}}^{(\text{ood})}$, descriptions for each class are generated. Given the high similarity between an ID class name and its corresponding OOD class names, a novel strategy is devised to generate distinctive descriptions for each ID and OOD class.

For each ID class name and its paired similar OOD class names, we prompt LLMs to generate the description $\mathbf{D}^{(\text{id})} = \{\mathbf{d}_1^{(\text{id})}, \dots, \mathbf{d}_c^{(\text{id})}\}$ that characterizes the ID class name concisely and precisely. For example, for the ID class name “sheep”, the generated descriptions might include “white wool”, “long neck and snout”. These descriptions may overlap with features of similar OOD classes, for example, the OOD class “deer” similar to ID class “sheep”, could also be described as “long neck and snout”. Next, we prompt LLMs to generate concise and precise descriptions for paired OOD class names. In Appendix A.2, we analyze the impact of common and specific features on OOD detection, demonstrating that only specific features can enhance OOD detection. To ensure these descriptions highlight unique properties and avoid overlapping with the ID class, the LLMs are explicitly instructed to focus on distinguishing features in their generated descriptions $\mathbf{D}^{(\text{ood})} = \bigcup_{i=1}^c \{\mathbf{d}_{i1}^{(\text{ood})}, \dots, \mathbf{d}_{ik}^{(\text{ood})}\}$. The prompt for description generation is given in Figure 3.

3.3 OOD Detection Score Designing

A novel method is proposed to compute the ID similarity score to complete the OOD detection task.

For any image \mathbf{x}_l and ID class name $\mathbf{y}_i^{(\text{id})}$, we utilize the OOD class names $\{\hat{\mathbf{y}}_{i1}^{(\text{ood})}, \dots, \hat{\mathbf{y}}_{ik}^{(\text{ood})}\}$ corresponding to $\mathbf{y}_i^{(\text{id})}$, the ID description $\mathbf{d}_i^{(\text{id})}$ corresponding to $\mathbf{y}_i^{(\text{id})}$, and the OOD descriptions $\{\mathbf{d}_{i1}^{(\text{ood})}, \dots, \mathbf{d}_{ik}^{(\text{ood})}\}$ to obtain the confidence score of image and ID class. Since there is a one-to-one correspondence between the class name and their descriptions, we first combine them using the following prompt to obtain an input text:

$$\mathbf{t} = \text{“a photo of \{CLASS_NAME\} with \{DESCRIPTION\}.”} \quad (1)$$

By respectively substituting *CLASS_NAME* and *DESCRIPTION* with the ID class name and its description, we obtain $\mathbf{t}_i^{(\text{id})}$. Similar operations are used to obtain $\{\mathbf{t}_{i1}^{(\text{ood})}, \dots, \mathbf{t}_{ik}^{(\text{ood})}\}$. Then, according to image \mathbf{x}_l , text $\mathbf{t}_i^{(\text{id})}$ and $\{\mathbf{t}_{i1}^{(\text{ood})}, \dots, \mathbf{t}_{ik}^{(\text{ood})}\}$, we can calculate features by using:

$$\begin{aligned} \mathbf{u}_l &= \phi(\mathbf{x}_l), \\ \mathbf{v}_i^{(\text{id})} &= \psi(\mathbf{t}_i^{(\text{id})}), \\ \forall j \in \{1, \dots, k\}, \mathbf{v}_{ij}^{(\text{ood})} &= \psi(\mathbf{t}_{ij}^{(\text{ood})}). \end{aligned}$$

Then, we can calculate the similarity by:

$$\begin{aligned} s_{il}^{(\text{id})} &= \frac{\mathbf{u}_l^\top \mathbf{v}_i^{(\text{id})}}{\|\mathbf{u}_l\| \|\mathbf{v}_i^{(\text{id})}\|}, \\ \forall j \in \{1, \dots, k\}, s_{ijl}^{(\text{ood})} &= \frac{\mathbf{u}_l^\top \mathbf{v}_{ij}^{(\text{ood})}}{\|\mathbf{u}_l\| \|\mathbf{v}_{ij}^{(\text{ood})}\|}. \end{aligned}$$

Based on the similarity $s_{il}^{(\text{id})}$ and $\{s_{ijl}^{(\text{ood})}\}_{j=1}^k$, a α -amended similarity strategy is proposed. Intuitively, when the model is more inclined to classify an image as belonging to an OOD class, the confidence in the ID class should decrease. Formally, an amended similarity is defined as:

$$\hat{s}_{il}^{(\text{id})} = \begin{cases} \alpha s_{il}^{(\text{id})} & \text{if } s_{il}^{(\text{id})} \leq \max_{j=1}^k \{s_{ijl}^{(\text{ood})}\}, \\ s_{il}^{(\text{id})} & \text{otherwise,} \end{cases}$$

where $0 < \alpha < 1$.

For now, we obtain the amended similarity score between an image and all ID class names. Similar to MCM [18], normalized confidence is used to determine whether an image belongs to an ID class. Specifically, the confidence is calculated using the following formula:

$$\text{sim}(\mathbf{x}_l, \mathbf{t}^{(\text{id})}) = \max_{i=1}^c \frac{e^{s_{il}^{(\text{id})}/\tau}}{\sum_{j=1}^c e^{s_{jl}^{(\text{id})}/\tau}},$$

where τ is the temperature coefficient. Then, if the confidence score is larger than a threshold parameter λ , \mathbf{x}_l is predicted as belonging to ID classes. Otherwise, \mathbf{x}_l belongs to OOD classes. The detailed algorithm is provided in Appendix A.3.

4 EXPERIMENTS

Table 1. Comparison of methods on different OOD datasets. The **black bold** indicates the best performance. \uparrow indicates larger values are better, while \downarrow indicates smaller values are better. All numbers in the table are expressed as percentages.

Methods	OOD Dataset									
	iNaturalist		SUN		Places		Textures		Average	
	AUROC \uparrow	FPR95 \downarrow	AUROC \uparrow	FPR95 \downarrow	AUROC \uparrow	FPR95 \downarrow	AUROC \uparrow	FPR95 \downarrow	AUROC \uparrow	FPR95 \downarrow
Requires training (or w. fine-tuning)										
MSP [3]	87.44	58.36	79.73	73.72	79.67	74.41	79.69	71.93	81.63	69.61
ODIN [28]	94.64	30.22	87.17	54.04	85.54	55.06	87.85	57.61	88.80	47.75
Energy [14]	95.33	26.12	92.66	35.97	91.41	39.87	86.76	57.61	91.54	39.89
GradNorm [29]	72.56	81.50	72.86	82.00	73.70	80.41	70.26	79.36	72.35	80.82
ViM [32]	93.16	32.19	87.19	54.01	83.75	60.67	87.18	53.94	87.82	50.20
KNN [30]	94.52	29.17	92.67	35.62	91.02	39.61	85.67	64.35	90.97	42.19
VOS [25]	94.62	28.99	92.57	36.88	91.23	38.39	86.33	61.02	91.19	41.32
NPOS [44]	96.19	16.58	90.44	43.77	89.44	45.27	88.80	46.12	91.22	37.93
ZOC [33]	86.09	87.30	81.20	81.51	83.39	73.06	76.46	98.90	81.79	85.19
CLIPN [19]	95.27	23.94	93.93	26.17	92.28	33.45	90.93	40.83	93.10	31.10
Zero-shot (w/o. fine-tuning)										
Mahalanobis [13]	55.89	99.33	59.94	99.41	65.96	98.54	64.23	98.46	61.50	98.94
Energy [14]	85.09	81.08	84.24	79.02	83.38	75.08	65.56	93.65	79.57	82.21
MCM [18]	94.59	32.20	92.25	38.80	90.31	46.20	86.12	58.50	90.82	43.93
Dai et al. [21]	95.54	22.88	92.60	34.29	89.87	41.63	87.71	52.02	91.43	37.71
EOE [15]	97.52	12.29	95.73	20.4	92.95	30.16	85.64	57.53	92.96	30.09
NegLabel [16]	99.49	1.91	95.49	20.53	91.64	35.59	90.22	43.56	94.21	25.40
Ours	98.59	6.03	96.52	18.72	93.13	28.86	92.22	39.15	95.12	23.19

4.1 Datasets and Metrics

Datasets We follow the dataset settings of MCM [18]. For the ID datasets, we use ImageNet-1k [37]. For the OOD datasets, we use iNaturalist [38], SUN [39], Places [40], and Texture [41]. Simultaneously, consistent with the settings of works like MCM [18], we use a subset of ImageNet-1k and the Waterbirds dataset [42] as ID datasets. For hard OOD detection, we employ another subset of ImageNet-1k and the Spurious OOD dataset [43] as OOD datasets.

Metrics Following the setting of prior researches [15, 16, 18, 19], we utilize two metrics: (1) the area under the ROC curve (AUROC), and (2) the false positive rate at 95% true positive rate (FPR95) for OOD samples.

4.2 Compared Methods

We compare our approach with the current SOTA OOD detection models, including both zero-shot and fine-tuning models. For the fine-tuning models, we compare MSP [3], ODIN [28], Energy [14], GradNorm [29], ViM [32], KNN [30],

VOS [25], NPOS [44], CLIPN [19], and ZOC [33]. For the zero-shot models, we compare MCM [18], as well as post-hoc methods implemented on the CLIP architecture, including Mahalanobis [13] and Energy [14] as additional baselines, and methods proposed by Dai et al. [21], NegLabel [16], and EOE [15], which supplement category names or descriptions with text. It is worth noting that CLIPN [19] use an additional large-scale dataset CC-3M [45] to pre-train the text encoder.

4.3 Implementation Details

We use CLIP [17] as the backbone of our framework, employing ViT-B/16 as the image encoder and masked self-attention Transformer as the text encoder. We adopt the pre-trained weights for CLIP provided by OpenAI. For LLMs, we utilize LLaMA-3-8b [35] for our research, with pre-trained weights sourced from Meta. In the experiments, unless otherwise specified, we use $k = 3$ to generate OOD classes corresponding to each ID class and set α to 0.8. We select the threshold value of λ when 95% of the ID samples are correctly classified and $T = 1$ as the temperature, following the standard practice [18, 46].

Table 2. Zero-shot OOD detection performance on hard OOD detection tasks.

Methods	ID:ImageNet-10		ID:ImageNet-20		ID:Waterbirds		Average	
	OOD:ImageNet-20		OOD:ImageNet-10		OOD:Spurious OOD			
	AUROC \uparrow	FPR95 \downarrow	AUROC \uparrow	FPR95 \downarrow	AUROC \uparrow	FPR95 \downarrow	AUROC \uparrow	FPR95 \downarrow
Mahalanobis [13]	90.71	51.46	90.41	37.5	99.55	2.21	93.56	30.39
Energy [14]	97.94	10.30	97.37	16.40	97.16	7.76	97.49	11.49
MCM [18]	98.71	5.00	98.09	12.91	93.30	14.45	96.70	11.12
Dai et al. [21]	98.77	4.20	98.26	9.24	98.62	4.56	98.55	6.00
EOE [15]	99.09	4.20	98.10	13.93	97.69	6.18	98.29	8.10
NegLabel [16]	98.86	5.10	98.81	4.60	94.67	9.50	97.45	6.40
Ours	99.32	1.10	99.23	1.40	99.09	4.30	99.21	2.27

4.4 Performance Comparison

OOD Detection on Large-Scale Datasets We use ImageNet-1k as the ID dataset and iNaturalist, SUN, Places, and Texture as the OOD datasets. Table 1 compares our approach with the latest SOTA methods, including both training-based and zero-shot inference methods. For each ID class, we set $k=3$, generating three OOD classes per ID. Our method achieves SOTA performance on the ImageNet-1k benchmark and surpasses a range of methods that employ fine-tuning for OOD detection, demonstrating the robust zero-shot OOD detection capabilities of CLIP. This is due to CLIP’s ability to match images with nuanced text descriptions. Additionally, the k OOD classes constructed for each

Table 3. Results after integrating our method with various scoring functions as baselines. The ID datasets are ImageNet-10, ImageNet-20, and Waterbirds, with corresponding OOD datasets being ImageNet-20, ImageNet-10, and Spurious OOD, respectively. "Average" represents the mean performance across these three datasets, and "Improvement" indicates the enhancement relative to the baseline.

Method	Average		Improvement	
	AUROC \uparrow	FPR95 \downarrow	AUROC \uparrow	FPR95 \downarrow
MCM	96.70	11.12	/	/
MCM _{our}	99.16	2.50	+2.46	-8.62
Energy	97.49	11.49	/	/
Energy _{our}	97.83	8.92	+0.34	-2.57
MaxLogit	97.67	10.84	/	/
MaxLogit _{our}	98.01	8.82	+0.34	-2.02

ID, even if they do not include the exact class names of the OOD samples encountered, provide an expanded feature space that facilitates matching OOD samples, thereby enhancing the model’s robustness. It is noteworthy that our method is slightly outperformed by NegLabel on the iNaturalist dataset. This is because NegLabel generates a large number of OOD class names, and the iNaturalist dataset contains a substantial number of plant species, among which these generated OOD class names are included.

OOD Detection on Hard OOD Datasets We extend our framework for hard OOD detection by categorizing it into two types: semantically hard OOD and spurious OOD. Semantically hard OOD refers to OOD samples that are semantically similar to the ID samples; for this, we use ImageNet-10 and ImageNet-20 as the ID and OOD datasets, respectively, and vice versa. Spurious OOD refers to OOD samples that have false correlations with the ID samples, such as the spurious correlation between habitats and bird species. The results of our hard OOD detection experiments are presented in Table 2. Our method improves the average FPR95 and AUROC by 3.73% and 0.66%, respectively, compared to the current best methods. Specifically, on the task with ImageNet-10 as ID and ImageNet-20 as OOD, our method achieves improvements of 3.10% in FPR95 and 0.29% in AUROC, and with ImageNet-20 as ID and ImageNet-10 as OOD, improvements of 3.20% in FPR95 and 0.42% in AUROC are observed. These results demonstrate the outstanding performance of our method in semantically hard OOD tasks.

4.5 Ablation Studies

Score Functions Our method adjusts confidence only for ID classes, enabling combination with various OOD detection scores. We denote pre-adjustment

Table 4. Impact of using different LLMs on results, consistent dataset settings as in Table 3. “A” represents the AUROC, “F” represents the FPR95.

Method	Average		Improve	
	A \uparrow	F \downarrow	A \uparrow	F \downarrow
MCM	96.70	11.12	/	/
LLaMA-3-8b	99.16	2.50	+2.46	-8.62
Claude2	99.03	3.21	+2.33	-7.91
GPT-4.0	99.06	3.00	+2.36	-8.12

Table 5. Impact of number of OOD classes on results, consistent dataset settings as in Table 3. k denotes the number of selected OOD classes.

Number	Average		Improve	
	A \uparrow	F \downarrow	A \uparrow	F \downarrow
MCM	96.70	11.12	/	/
$k=1$	98.99	2.92	+2.29	-8.20
$k=2$	99.00	2.53	+2.30	-8.59
$k=3$	99.27	2.20	+2.57	-8.92
$k=4$	99.06	2.43	+2.36	-8.69
$k=5$	99.08	2.74	+2.38	-8.38

scores as MCM, Energy, and MaxLogit, and post-adjustment scores as MCM_{our} , $\text{Energy}_{\text{our}}$, and $\text{MaxLogit}_{\text{our}}$. This validates MCM score’s impact on our ablation study. After ID confidence adjustment, OOD detection with MCM, Energy, and MaxLogit improves performance across three datasets (Table 3). MCM yields AUROC and FPR95 gains of 2.46% and 8.62%, respectively, due to our scaling factor $\alpha = 0.8$, which amplifies ID-OOD score differences.

The Choice of LLMs We conduct experiments using various LLMs to comprehensively assess the effectiveness of descriptors generated by different LLMs. Specifically, we utilize LLaMA-3-8b, ChatGPT-4, and Claude 2 for descriptor generation. The average results across three datasets, shown in Table 4, indicate that using different LLMs achieved better performance compared to the baseline MCM. Additionally, LLaMA-3-8b outperforms Claude2 and GPT-4.0 in both AUROC and FPR95 metrics, demonstrating the generalizability and robustness of our method. This can be attributed to the fact that LLaMA-3-8b excels at generating short, task-specific text, which benefits OOD detection by providing focused descriptions [47]. In contrast, GPT-4, while powerful in broader tasks, may produce more verbose responses that could introduce noise into similarity comparisons. Therefore, the performance differences between the two are likely due to factors such as the relevance and specificity of generated OOD class names, the precision of descriptive terms, and prompt interpretation.

Fine-grained Textual Features To further validate the effectiveness of descriptors, we conduct an ablation where descriptors are simplified to only use generated hard OOD classes for inference, altering the template to “a photo of {Class_Name}” to assess the efficacy of textual features. We refer to this as “Label only” in Figure 5. Additionally, to validate the effectiveness of our descriptors in distinguishing OOD from current ID samples, we designed different prompts for verification. Specifically, we modify the question in Figure 3 to directly ask, “Please describe the visual characteristics of {OOD_CLASS}”, “What are the visual features similar to {OOD_CLASS} and {ID_CLASS}?”

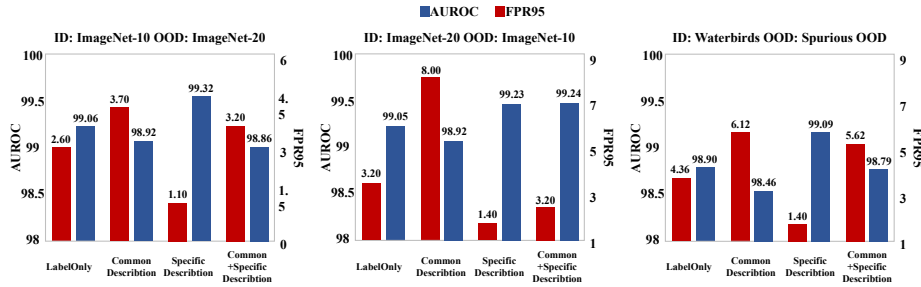


Fig. 5. Impact of using different prompts to generate various types of descriptors with LLMs.

to obtain descriptions that include both common and specific OOD features, as well as descriptions with only specific OOD features. In Figure 5, these are represented as “Common+Specific description” and “Common description”, respectively. “Specific description” represents the primary method of this paper, obtaining unique features that distinguish OOD classes from ID classes.

The results indicate that even when only category names are used to provide textual information (“Label only”), our method still outperforms the baseline MCM. However, when the textual description includes a significant amount of ID features (“Common description”), the performance of OOD detection significantly decreases, with the most notable decline observed on the ImageNet-20 dataset, where FPR95 and AUROC drop by 4.80% and 0.52%, respectively. When the OOD descriptors include only the unique features that distinguish OOD classes from ID classes (“Specific description”), our method achieves the best performance across all three datasets. When descriptors include both types of features (“Common+Specific description”), the results on various datasets are better than those with only the common description, but still inferior to those with only the specific description. This validates that common features shared between OOD and ID classes are detrimental to OOD detection.

4.6 Hyperparameter Tune-up

Number of OOD Class Labels We investigate the impact of the number of generated OOD classes k per ID class on performance. As shown in Table 5, performance improves initially and then declines as k increases, with the best results at $k = 3$. The initial gain stems from the expanded textual space, enhancing the model’s capacity to separate ID and OOD samples. However, larger k increases the likelihood of generating semantically similar OOD descriptions, causing feature overlap with ID or existing OOD classes. This overlap can be attributed to the inherent characteristics of LLMs, which may generate similar descriptions for semantically related concepts. For example, OOD descriptions for “cat” may include “kitty” and “kitten”, blurring decision boundaries and impairing performance when $k > 3$.

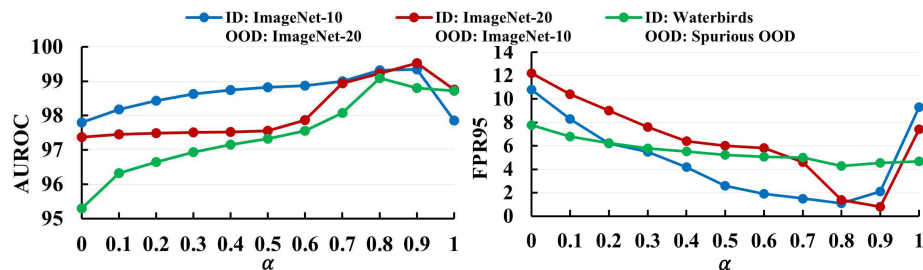


Fig. 6. Performance Evaluation of the Hyperparameter α for ID Confidence Correction. The upper and lower graphs respectively represent the score trends for FPR95 and AUROC.

ID Confidence Calibration We explore the impact of the ID confidence calibration coefficient, the hyperparameter α , on the results. We conduct experiments on the ImageNet-10, ImageNet-20, and Waterbirds datasets, and the results are shown in Figure 6. We observe that when α is set to extreme values of 0 and 1, and the performance of OOD detection significantly decreases. When α is 0, setting the confidence directly to 0 leads to some ID samples being incorrectly classified as OOD, reducing robustness. Conversely, when α is 1, not adjusting the confidence negates the model’s effectiveness. However, when α is between 0.7 and 0.9, the model performs well across all three datasets, indicating that our model is not sensitive to the α parameter.

5 CONCLUSION

In this paper, we proposed a novel zero-shot OOD detection approach leveraging LLMs to enhance the textual feature extraction for both ID and OOD classes. Specifically, we utilized LLMs to generate descriptive terms for ID data and challenging names and detailed descriptions for OOD classes. This method enabled VLMs to focus on relevant regions of images, significantly improving zero-shot OOD detection performance. We adjust the confidence scores of ID samples based on the confidence of generated OOD classes and introduce a new scoring method for detecting OOD samples. Extensive experiments on multiple OOD detection benchmarks demonstrated the superiority of our approach over SOTA methods. Our results highlight the potential of leveraging LLMs for nuanced textual feature generation to advance OOD detection. Future work could explore the integration of additional knowledge to further enhance the robustness and accuracy of our framework.

Acknowledgments. This work is supported by the National Key RD Program of China (2022YFF0712100), NSFC (62276131), Natural Science Foundation of Jiangsu Province of China under Grant (BK20240081).

References

1. Abhijit Bendale, Terrance E. Boult, Towards Open World Recognition, in: CVPR, 2015.
2. Bendale, Abhijit and Boult et al., Towards open set deep networks, in: CVPR, 2016.
3. Dan Hendrycks, Kevin Gimpel, A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks, in: ICLR, 2017.
4. Jingkang Yang, Kaiyang Zhou and Yixuan Li et al., Generalized Out-of-Distribution Detection: A Survey, *International Journal of Computer Vision* (2024).
5. Jingyang Zhang, Jingkang Yang and Pengyun Wang et al., OpenOOD v1.5: Enhanced Benchmark for Out-of-Distribution Detection, arXiv, abs/2306.09301 (2023).
6. Gao Huang, Zhuang Liu and Laurens van der Maaten et al., Densely Connected Convolutional Networks, in: CVPR, 2017.
7. Di Feng, Ali Harakeh and Steven L. Waslander et al., A Review and Comparative Study on Probabilistic Object Detection in Autonomous Driving, *Trans. Intell. Transp. Syst.* (2022).
8. Li Chen, Penghao Wu and Kashyap Chitta et al., End-to-end Autonomous Driving: Challenges and Frontiers, *Transactions on Pattern Analysis and Machine Intelligence* (2024).
9. Igor Kononenko, Machine learning for medical diagnosis: history, state of the art and perspective, *Artif. Intell. Medicine* (2001).
10. Yen-Chang Hsu, Yilin Shen and Hongxia Jin et al., Generalized ODIN: Detecting Out-of-Distribution Image Without Learning From Out-of-Distribution Data, in: CVPR, 2020
11. Haoran Wang, Weitang Liu and Alex Bocchieri et al., Can multi-label classification networks know what they don't know? in: NIPS, 2021.
12. Vikash Sehrawag, Mung Chiang and Prateek Mittal et al., SSD: A Unified Framework for Self-Supervised Outlier Detection, in: ICLR, 2021.
13. Kimin Lee, Kibok Lee and Honglak Lee et al., A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks, in: NIPS, 2018.
14. Weitang Liu, Xiaoyun Wang and John D. Owens et al., Energy-based Out-of-distribution Detection, in: NIPS, 2020.
15. Chentao Cao, Zhun Zhong and Zhanke Zhou et al., Envisioning Outlier Exposure by Large Language Models for Out-of-Distribution Detection, in: ICML, 2024.
16. Xue Jiang, Feng Liu and Zhen Fang et al., Negative Label Guided OOD Detection with Pretrained Vision-Language Models, in: ICLR, 2024.
17. Alec Radford, Jong Wook Kim and Chris Hallacy et al., Learning Transferable Visual Models From Natural Language Supervision, in: ICML, 2021.
18. Yifei Ming, Ziyang Cai and Jiuxiang Gu et al., Delving into Out-of-Distribution Detection with Vision-Language Representations, in: NIPS, 2022.
19. Hualiang Wang, Yi Li and Huifeng Yao et al., CLIPN for Zero-Shot OOD Detection: Teaching CLIP to Say No, in: ICCV, 2023.
20. Fellbaum, Christiane, *WordNet: An electronic lexical database*, MIT press (1998).
21. Yi Dai, Hao Lang and Kaisheng Zeng et al., Exploring Large Language Models for Multi-Modal Out-of-Distribution Detection, in: EMNLP, 2023.
22. Fabio Petroni, Tim Rocktäschel and Sebastian Riedel et al., Language Models as Knowledge Bases? in: EMNLP, 2019.

23. Jihoon Tack, Sangwoo Mo and Jongheon Jeong et al., CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances, in: NIPS, 2020.
24. Rui Huang, Yixuan Li, MOS: Towards Scaling Out-of-Distribution Detection for Large Semantic Space, in: CVPR, 2021.
25. Xuefeng Du, Xin Wang and Gabriel Gozum et al., Unknown-Aware Object Detection: Learning What You Don't Know from Videos in the Wild, in: CVPR, 2022.
26. Hongxin Wei, Renchunzi Xie and Hao Cheng et al., Mitigating Neural Network Overconfidence with Logit Normalization, in: ICML, 2022.
27. Yifei Ming, Yiyu Sun and Ousmane Dia et al., CIDER: Exploiting Hyperspherical Embeddings for Out-of-Distribution Detection, arXiv, abs/2203.04450 (2022).
28. Xue Jiang, Feng Liu and Zhen Fang et al., Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks, in: ICLR, 2018.
29. Rui Huang, Andrew Geng and Yixuan Li et al., On the Importance of Gradients for Detecting Distributional Shifts in the Wild, in: NIPS, 2021
30. Yiyu Sun, Yifei Ming and Xiaojin Zhu et al., Out-of-Distribution Detection with Deep Nearest Neighbors, in: ICML, 2022.
31. LeCun, Yann and Chopra et al., A tutorial on energy-based learning, Predicting structured data (2006).
32. Haoqi Wang, Zhizhong Li and Litong Feng et al., ViM: Out-Of-Distribution with Virtual-logit Matching, in: CVPR, 2022.
33. Sepideh Esmaeilpour, Bing Liu and Eric Robertson et al., Zero-Shot Out-of-Distribution Detection Based on the Pre-trained Model CLIP, in: AAAI, 2022.
34. Tom B. Brown, Benjamin Mann and Nick Ryder et al., Language Models are Few-Shot Learners, in: NIPS, 2020.
35. Hugo Touvron, Louis Martin and Kevin Stone et al., Llama 2: Open Foundation and Fine-Tuned Chat Models, arXiv, abs/2307.09288 (2023).
36. OpenAI, GPT-4 Technical Report, arXiv, abs/2303.08774 (2023).
37. Jia Deng, Wei Dong and Richard Socher et al., ImageNet: A large-scale hierarchical image database, in: CVPR, 2009.
38. Grant Van Horn, Oisin Mac Aodha and Yang Song et al., The INaturalist Species Classification and Detection Dataset, in: CVPR, 2018.
39. Jianxiong Xiao, James Hays and Krista A. Ehinger et al., SUN database: Large-scale scene recognition from abbey to zoo, in: CVPR, 2010.
40. Bolei Zhou, Àgata Lapedriza and Aditya Khosla et al., Places: A 10 Million Image Database for Scene Recognition, Trans. Pattern Anal. Mach. Intell. (2018).
41. Mircea Cimpoi, Subhransu Maji and Iasonas Kokkinos et al., Describing Textures in the Wild, in: CVPR, 2014.
42. Shiori Sagawa, Pang Wei Koh and Tatsunori B. Hashimoto et al., Distributionally Robust Neural Networks for Group Shifts: On the Importance of Regularization for Worst-Case Generalization, arXiv, abs/1911.08731 (2019).
43. Yifei Ming, Hang Yin and Yixuan Li et al., On the Impact of Spurious Correlation for Out-of-Distribution Detection, in: AAAI, 2022.
44. Leitian Tao, Xuefeng Du and Jerry Zhu et al., Non-parametric Outlier Synthesis, in: ICLR, 2023.
45. Piyush Sharma, Nan Ding and Sebastian Goodman et al., Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning, in: ACL, 2018.
46. Yichen Bai, Zongbo Han and Bing Cao et al., ID-like Prompt Learning for Few-Shot Out-of-Distribution Detection, in: CVPR, 2024.

47. Zongxi Li, Xianming Li and Yuzhang Liu et al., Label Supervised LLaMA Fine-tuning, arXiv, abs/2310.01208 (2023).