

M3OOD: Automatic Selection of Multimodal OOD Detectors

Yuehan Qin, Li Li, Defu Cao, Tiankai Yang, Jiate Li, Yue Zhao

University of Southern California
{yuehanqi, lli15753, defucao, tiankaiy, jiateli, yzhao010}@usc.edu

Abstract

Out-of-distribution (OOD) robustness is a critical challenge for modern machine learning systems, particularly as they increasingly operate in multimodal settings involving inputs like video, audio, and sensor data. Currently, many OOD detection methods have been proposed, each with different designs targeting various distribution shifts. A single OOD detector may not prevail across all the scenarios; therefore, how can we automatically select an ideal OOD detection model for different distribution shifts? Due to the inherent unsupervised nature of the OOD detection task, it is difficult to predict model performance and find a universally Best model. Also, systematically comparing models on the new unseen data is costly or even impractical. To address this challenge, we introduce M3OOD, a meta-learning-based framework for OOD detector selection in multimodal settings. Meta learning offers a solution by learning from historical model behaviors, enabling rapid adaptation to new data distribution shifts with minimal supervision. Our approach combines multimodal embeddings with handcrafted meta-features that capture distributional and cross-modal characteristics to represent datasets. By leveraging historical performance across diverse multimodal benchmarks, M3OOD can recommend suitable detectors for a new data distribution shift. Experimental evaluation demonstrates that M3OOD consistently outperforms 10 competitive baselines across 12 test scenarios with minimal computational overhead.

1 Introduction

Out-of-distribution (OOD) detection aims to identify samples that differ markedly from the distribution of the training data. For multimodal machine learning (ML) systems handling diverse modality inputs like vision, text, and audio, this capability is essential for maintaining robustness (Yang et al. 2021; Song et al. 2024, 2025b; Hu et al. 2024; Wang* et al. 2024). and is particularly important in high-risk domains such as autonomous driving (Li et al. 2024a), medical diagnostics (Ulmer, Meijerink, and Cinà 2020), and other applications (Xu et al. 2025a,b,c; Sun et al. 2023; Song et al. 2022, 2025a; Shi et al. 2025; Song et al. 2023; Li et al. 2023). As ML systems are increasingly adopted in multimodal settings (Radford et al. 2021; Zhang et al. 2023; Li et al. 2024b, 2025a,c), researchers have begun to explore specialized benchmarks and frameworks for multimodal OOD detection (Dong et al. 2024a; Li et al. 2025b).

However, although a broad range of OOD detection methods have been proposed, each tuned to capture particular characteristics of data distributions, there is still no systematic approach for selecting the most suitable OOD detector under multimodal settings. This challenge stems from the inherently unsupervised nature of the OOD detection, which makes predicting model performance and identifying a universally optimal model challenging. Given that each OOD detector is based on distinct assumptions and methodological choices, selecting a single model for all distributions is ineffective, and exhaustively training one per case is infeasible. This is even compounded by the fact that cross-modal alignment inconsistencies and modality-specific distribution shifts can cause methods that perform well on individual modalities to fail when modalities are combined, consistent with the no-free-lunch theorem (Wolpert and Macready 1997). Furthermore, conducting systematic model comparisons on the new data distribution shifts can be prohibitively expensive or unfeasible. As a result, we require an automated framework that can identify the most appropriate OOD detector without expensive evaluations.

Present Work. In the related field of outlier detection (OD), unsupervised OD method selection has advanced through meta-learning as in MetaOD (Zhao, Rossi, and Akoglu 2021), ELECT (Zhao, Zhang, and Akoglu 2022), ADGym (Jiang et al. 2024), and HyPer (Ding, Zhao, and Akoglu 2024). However, these approaches cannot be directly applied to OOD detection due to differences in task structure and data complexity. OOD detection is trained only on in-distribution (ID) data to identify samples from entirely different distributions, often across multiple datasets. It often deals with high-dimensional visual data, in contrast to OD, which is commonly applied to time-series or tabular datasets. Extending from unimodal OOD data (Qin et al. 2025) to multimodal inputs introduces additional challenges: adapting embeddings to handle temporal sequences and spatial information, maintaining efficiency as dataset size grows, and creating unified representations for heterogeneous data types. More critically, designing effective multimodal meta-features for model selection becomes more difficult due to the complex interactions between modalities and the lack of clear guidelines for capturing cross-modal relationships in feature representations. Differences in data representation, distributional characteristics, and detection

behavior across modalities require a specialized approach for multimodal OOD detection model selection.

Our Work. To address the aforementioned challenges, we present M3OOD (Fig. 1), the *first* model selection approach for OOD detection in multimodal settings, based on meta-learning. We show that by combining multimodal model embeddings with handcrafted meta-features capturing distributional and modality-specific properties, our meta-learning based approach can unify representations for video and optical flow. Alterations to a single modality can be reflected in the meta-embeddings, enabling the model to adapt its detector selection accordingly. The central idea is that an OOD detector that performs well on previous datasets with similar properties is likely to generalize well to new datasets. During the meta-training phase, we evaluate a pool of OOD detection methods across a wide range of carefully curated datasets spanning different modalities, including videos and spectrograms, to build a performance profile under varied conditions. When a new multimodal dataset is introduced, we utilize the knowledge accumulated from historical datasets to recommend an appropriate OOD detection method. This selection process is guided by estimating the similarity between the new dataset and those seen during meta-training. Our main contributions are:

- **First Multimodal OOD Detection Model Selection Framework.** To our best knowledge, we introduce the first meta-learning-based framework for zero-shot multimodal OOD detection model selection.
- **Specialized Multimodal Embeddings.** We leverage multimodal features to quantify similarity among OOD detection tasks, enhancing detector selection through better characterization of cross-modal OOD properties.
- **Superior Performance.** M3OOD outperforms eleven model selection methods and unsupervised meta-learners across twelve test data pairs, yielding statistically significant ranking improvements with efficient runtime.
- **Open-Source Release.** We provide the testbed and code at <https://github.com/yqin43/M3OOD>.

2 Related Work

2.1 Multimodal OOD Detection

Multimodal OOD detection has gained attention in recent work, particularly for vision-language architectures (Ming et al. 2022; Wang et al. 2023). Maximum Concept Matching (MCM) (Ming et al. 2022) leverages alignment between visual features and textual concept representations to generate OOD scores. CLIPN (Wang et al. 2023) enhances the CLIP architecture through contrasting prompt techniques that strengthen the distinction between ID and OOD data. Furthermore, comprehensive multimodal benchmarks incorporating video, optical flow, and audio modalities have been developed (Dong et al. 2024b; Li et al. 2025b), highlighting cross-modal prediction inconsistencies in OOD scenarios.

2.2 Unsupervised OOD Detector Selection

A core difficulty in OOD detection stems from the inherent unknowns: the characteristics of OOD samples and the prop-

erties of their underlying distributions remain hidden during the training phase (Hendrycks, Mazeika, and Dietterich 2019; Liang, Li, and Srikant 2018). This constraint forces model selection to operate in an unsupervised manner, as labeled OOD samples are unavailable for evaluating and comparing detector performance (Lee et al. 2018a). In real-world deployment scenarios, OOD detectors must handle test inputs spanning diverse and previously unseen distributional patterns (Yang et al. 2021), making unsupervised model selection approaches particularly advantageous for OOD detection applications (Liu et al. 2020a). While recent work has explored instance-specific selection of detector ensembles (Xue et al. 2024), such methods differ from our objective of identifying a single optimal detector for each dataset.

Many existing approaches to OOD detector selection rely on trial-and-error or empirical heuristics. A straightforward strategy is to default to widely used detectors such as Maximum Softmax Probability (Hendrycks and Gimpel 2017) or ODIN (Liang, Li, and Srikant 2017). Other simple methods include using the confidence scores from ID data as proxies for selecting OOD detectors. However, these heuristic strategies often yield suboptimal performance due to neural networks’ tendency toward overconfident OOD predictions (Hendrycks and Gimpel 2017). Another line of work explores similarity-based techniques, where model selection is guided by the resemblance between datasets or their clustering, a strategy previously applied in algorithm recommendation systems (Kadioglu et al. 2010; Nikolić, Marić, and Janičić 2013; Xu et al. 2012; Misir and Sebag 2017). We thus include these methods as baselines in our study.

2.3 Embedding-based Representations for Meta-learning

In meta-learning, effective data representation is essential for capturing dataset or task similarity, and embeddings serve as a key mechanism for this purpose. Traditionally, computational meta-features such as dataset statistics and model-independent properties have been widely used to represent data in meta-learning frameworks (Vanschoren 2018). More recently, advanced learning-based representations which aim to learn embeddings from data directly have emerged, including methods like dataset2vec (Jomaa, Schmidt-Thieme, and Grabocka 2021) and HyPer (Ding, Zhao, and Akoglu 2024). In parallel, language model and multimodal embeddings have been increasingly employed to encode dataset characteristics, offering a semantic-rich alternative that supports deeper model understanding (Drori et al. 2019; Fang et al. 2024; Qin et al. 2025). To leverage the strengths of both approaches, we combine handcrafted meta-features that capture distributional and multimodality-specific characteristics with SlowFast-generated embeddings for comprehensive multimodal dataset representation.

3 Methodology

3.1 Preliminaries on OOD Detection

Consider a training dataset $\mathbf{X}_{\text{train}}$ sampled from an in-distribution $\mathcal{P}_{\text{in}}$, such that $\mathbf{X}_{\text{train}} \sim \mathcal{P}_{\text{in}}$. Typically, a model

is trained solely on ID data to learn a task such as classification. At test time, we consider a dataset $\mathbf{X}_{\text{test}}$ that may contain both ID samples and samples from unknown distributions (OOD samples). The goal of OOD detection is to build a classifier G that decides for each test input $x \in \mathbf{X}_{\text{test}}$ whether it originates from $\mathcal{P}_{\text{in}}$:

$$G(x) = \begin{cases} ID & \text{if } x \in \mathcal{P}_{\text{in}}, \\ OOD & \text{if } x \notin \mathcal{P}_{\text{in}}. \end{cases}$$

When evaluating an OOD detector on $\mathbf{X}_{\text{test}}$, both ID and OOD test data are present to assess the detector’s capability to accurately distinguish between known and unknown data samples. Accordingly, in this study, we use dataset pairs $D = \{\mathbf{X}_{\text{train}}, \mathbf{X}_{\text{test}}\}$, where $\mathbf{X}_{\text{train}}$ contains only ID samples and $\mathbf{X}_{\text{test}}$ contains a mixture of ID and OOD samples, allowing us to assess how well the detector separates known from unknown data. When the input data includes multiple modalities, we extend this framework to multimodal OOD detection, which we define as follows:

MultiModal OOD Detection Each training sample x_i contains K distinct modalities, expressed as $x_i = \{x_i^k \mid k = 1, \dots, K\}$. Information from all these modalities is integrated to generate the final prediction by taking the combined embeddings from all modalities and outputs a score s . s may represent a probability, a confidence score, an energy value, or any other scalar used by the detector. Let $\psi(\cdot)$ be a feature extractor that maps an input x_i to an embedding E , and let $h(\cdot)$ be a scoring function or classifier that maps this embedding to an output score s . The overall output used for OOD detection is:

$$s = h(\psi(x_i)) = h([\psi(x_1), \dots, \psi(x_K)]) = h([E_1, \dots, E_K]),$$

A sample with score above the threshold η is classified as ID; otherwise, it is classified as OOD:

$$G(x) = \begin{cases} ID & \text{if } s \geq \eta, \\ OOD & \text{if } s < \eta. \end{cases}$$

3.2 Problem Statement and Framework Overview

Given a new and previously unseen pair of datasets $D_{\text{new}} = \{\mathbf{X}_{\text{train}}^{\text{new}}, \mathbf{X}_{\text{test}}^{\text{new}}\}$ as input, our objective is to choose the best candidate OOD detection model $M \in \mathcal{M}$ *without* conducting test-time model evaluations, where we have no ground truth labels $\mathbf{y}_{\text{test}}^{\text{new}}$ for evaluation. In this work, we adopt a meta-learning approach to transfer performance knowledge from previously encountered tasks to the new OOD detection setting. Meta-learning (Vanschoren 2018), often referred to as “learning to learn”, involves training across a range of historical or meta-level tasks, enabling the algorithm to generalize effectively to novel tasks. The underlying intuition is that an OOD detector that performs well on past datasets with similar characteristics is also likely to perform well on a new, related dataset. This strategy is especially beneficial in situations where model evaluation is impractical or costly due to the absence of ground truth labels or the need for quick deployment.

The proposed meta-learner, M3OOD, relies on:

(1) A collection of n historical (i.e., meta-train) OOD detection dataset pairs, $\mathcal{D}_{\text{train}} = \{D_1, \dots, D_n\}$ with ground truth labels, i.e., $D = \{\mathbf{X}_{\text{train}}, (\mathbf{X}_{\text{test}}, \mathbf{y}_{\text{test}})\}$ where $\mathbf{X} = [\mathbf{X}^1, \dots, \mathbf{X}^K]$ denotes K -modality data.

(2) Historical performance $\mathbf{P}$ of the pre-set model set $\mathcal{M} = \{M_1, \dots, M_m\}$ (with m models), on the meta-train datasets. We refer to $\mathbf{P} \in \mathbb{R}^{n \times m}$ as the performance matrix, where $\mathbf{P}_{i,j}$ corresponds to the j -th model M_j ’s performance on the i -th meta-train dataset pair D_i .

Our method includes two stages: (i) an offline training stage, where a model is trained to capture how different OOD detection models perform across a set of historical datasets $\mathcal{D}_{\text{train}}$, and (ii) an online stage, where this prior information is used to select an appropriate model for a new test dataset D_{new} . Fig. 1 outlines the workflow and key elements of M3OOD, with the offline training phase shown at top and the online model selection stage shown at the bottom.

3.3 Offline Meta-Training

In the offline training stage (Fig. 1 top), we construct embeddings that represent each combination of a dataset pair $\mathcal{D}_{\text{train}}$ and a method $\mathcal{M}$, and then learn a latent function that maps these embeddings to their corresponding performance values $\mathbf{P}$. By capturing this relationship, the meta-learner can generalize to new, unseen datasets and identify the method expected to perform best based on the learned mapping $\mathcal{D}_{\text{train}}, \mathcal{M} \rightarrow \mathbf{P}$. Note this process is supervised.

To predict the performance of the *candidate model* on a *new dataset pair*, we propose training a meta-predictor as a regression problem. The input to the meta-predictor consists of $E_i^{\text{meta}}, E_j^{\text{model}}$, corresponding to the embedding of the i -th dataset pair and the embedding of the j -th OOD detector. Dataset embedding of dataset pair $\mathcal{D}$ is denoted as $E^{\text{data}} = \psi(\mathcal{D})$, and method embedding extracted from $\mathcal{M}$ is denoted as $E^{\text{model}} = \phi(\mathcal{M}, M)$; we provide more details below on the embedding generation in the following session. The meta-train process can be formulated as:

$$f : \mathcal{H} \times \mathcal{G} \rightarrow \mathbb{R}^+$$

$$\text{where } \mathcal{H} = \{[E_i^{\text{video}}, E_i^{\text{flow}}] \mid i \in \{1, \dots, n\}\}$$

$$\mathcal{G} = \{E_j^{\text{model}} \mid j \in \{1, \dots, m\}\}$$

$$f(E_i^{\text{dataset}}, E_j^{\text{model}}) = \hat{P}_{i,j} \quad \forall (i, j) \in [n] \times [m] \quad (1)$$

Our goal is to train the meta-predictor f^1 to map the characteristics of the datasets and the OOD detectors to their corresponding performance ranking across all historical dataset pairs. Specifically, the meta-train objective is as follows:

$$\min_f \mathbb{E}_{i \sim \mathcal{N}, j \sim \mathcal{M}} [\mathcal{L}(f(\psi(D_i), \phi(\mathcal{M}, M_j)), P_{i,j})] \quad (2)$$

3.4 Data and Model Embeddings

A central component of our meta-learning framework involves extracting meta-features that characterize the essen-

¹The format of f can be any regression models; in this work, we use an XGBoost (Chen and Guestrin 2016) model due to its balance of simplicity and expressiveness, as well as strong feature selection characteristic.

Category	OOD Detection Model
Probability-based	MSP (Hendrycks and Gimpel 2017)
	GEN (Liu, Lochman, and Zach 2023)
Logit-based	MaxLogit (Hendrycks et al. 2022)
	EnergyBased (Liu et al. 2020b)
Feature-based	Mahalanobis (Lee et al. 2018b)
	ViM (Wang et al. 2022)
	kNN (Cover and Hart 1967)
Activation Pruning	ReAct (Sun, Guo, and Li 2021)
	ASH (Djurisic et al. 2023)

Table 1: OOD detectors considered for model selection.

$$M^* := \arg \max_{M_j \in \mathcal{M}} \hat{\mathbf{P}}_{\text{new},j}, \quad \text{where} \quad \hat{\mathbf{P}}_{\text{new},j} = f(E_{\text{new}}^{\text{meta}}, E_j^{\text{model}}) \quad (3)$$

For a new dataset pair, the trained predictor f is used to estimate the relative performance of various OOD detectors. Based on these predictions, the method ranked highest is selected², as indicated in Eq. (3). Notably, this approach is *zero-shot*, meaning it does not involve any model training on the test data.

4 Experiments

Our experiments address the following research questions (RQ): **RQ1**: How effective is the proposed M3OOD in unsupervised OOD detector selection compared to leading baselines? **RQ2**: How do different design choices affect the performance of M3OOD? **RQ3**: How much time overhead/saving M3OOD introduces to multimodal OOD detection?

The model set $\mathcal{M}$. We construct $\mathcal{M}$ with 9 popular OOD detectors (Tab. 1) spanning diverse detection methods.

Datasets. In real-world settings, OOD data often differ from ID data not only in semantics but also in domain. To better reflect such situation, we design the dataset to include two types of distribution shifts: Far-OOD and Near-OOD. In the Far-OOD setting, we treat a full dataset as ID and use other datasets with related tasks but no overlapping categories as OOD. This introduces both semantic and domain shifts between ID and OOD samples. In the Near-OOD setting, we split the categories within a single dataset into two disjoint groups: one used as ID and the other as OOD. In this case, ID and OOD samples share the same underlying distribution, differing only in semantics. We use five action recognition datasets (EPIC-Kitchens (Munro and Damen 2020), HAC (Dong et al. 2023), HMDB51 (Kuehne et al. 2011), UCF101 (Soomro, Zamir, and Shah 2012), and Kinetics-600 (Carreira et al. 2018)), The Near-OOD and Far-OOD setup details are in Supplementary Material §4.

Train-test Split. We split the train-test sets as shown in Tab. 2 in the meta-training stage. Each row corresponds to a different meta-train/meta-test split. For example, in the first row, the meta-train set includes HMDB and Kinetics (Near-OOD and Far-OOD), and the corresponding meta-test set includes UCF and EPIC (Near-OOD). This setup ensures that

²Although selecting the top- k methods for ensemble use is possible, this work focuses on top-1 selection.

the meta-predictor is trained on diverse OOD conditions and evaluated on unseen datasets, allowing us to assess its ability to generalize OOD detector selection across both semantic similarity and distributional shifts.

Hardware. All models are implemented on the Multi-OOD codebase (Dong et al. 2024b) and run on a multi-NVIDIA RTX 6000 Ada workstation.

Training the meta-predictor f (see details in previous Offline Meta-Training section). In this work, we use an XGBoost (Chen and Guestrin 2016) model as f due to its simplicity and expressiveness.

Evaluation. To evaluate the performance of M3OOD against the baselines, we compare the rank of performance of the OOD detector selected by each method among all candidates. We use Area Under the ROC Curve (AUC-ROC) as the evaluation metric³, and visualize the results using boxplots and a rank diagram that reports the average rank across all dataset pairs. A rank of 1 indicates the best-performing selection, 11 is the worst (10 baselines plus M3OOD). For statistical comparison, we apply the pairwise Wilcoxon rank-sum test across dataset pairs with a significance threshold of $p < 0.05$.

Model Selection Baselines. We select the baselines based on the literature in meta-learning for unsupervised model selection (Zhao, Rossi, and Akoglu 2021; Zhao, Zhang, and Akoglu 2022; Jiang et al. 2024; Park et al. 2023), grouped into four categories, as shown in Tab. 4. Details of the model selection baselines are in Supplementary Material §3.

5 Results

In Fig. 2, we report the distribution of the true ranks of the top-1 OOD detector selected by each model selection method across the test data pairs. Also, we include the overall average-rank diagram in Fig. 3, which displays the mean performance rank of the OOD detector selected by each algorithm. To compare two model selection algorithms (e.g., ours with a baseline), we perform Wilcoxon rank test on the rank of the top-1 models selected by our method and the baseline method, as shown in Fig. 5. We summarize the main findings as below:

1. M3OOD outperforms all baselines. Fig. 2 demonstrates that M3OOD achieves stable, high-quality performance with minimal variance. It maintains the highest average ranking among all the 10 baseline methods that span from random or fixed selection to optimization or learning-based methods (Fig. 3). Additionally, Tab. 5 shows that most performance gains are statistically significant. This consistent pattern of results indicates that M3OOD effectively handles complex datasets while maintaining stable. We attribute this effectiveness to the integration of a meta-learning approach with our designed multimodal dataset embeddings.

2. Meta-learner perform better than other baselines. Meta-learners (M3OOD, AS, ALORS) significantly outperform single outlier detection methods and ME that averages all the model performances. Meanwhile, optimization-based meta learners (i.e. M3OOD, ALORS) demonstrate relatively

³Other metrics can be used at interest.

Train	Test
HMDB, Kinetics, HMDB-Far-OOD, Kinetics-Far-OOD	UCF, EPIC
UCF, EPIC, Kinetics, Kinetics-Far-OOD	HMDB, HMDB-Far-OOD
HMDB, UCF, EPIC, HMDB-Far-OOD	Kinetics, Kinetics-Far-OOD

Table 2: Meta-train train/test split (datasets without the “-Far-OOD” suffix are Near-OOD).

Table 3: Various OOD detectors’ performances on Near-OOD and Far-OOD dataset pairs. We highlight the selected OOD method for each dataset pair in the test set in bold.

OOD Dataset	HMDB51	UCF101	EPIC	Kinetics-600
Near-OOD				
MSP	87.74	95.73	67.59	76.16
Energy	87.46	96.06	68.29	75.49
MaxLogit	87.75	96.02	68.29	75.98
Mahalanobis	85.28	97.14	42.99	35.83
ReAct	87.09	95.85	65.89	73.80
ASH	87.16	94.02	67.92	76.16
GEN	87.49	95.64	68.52	75.33
KNN	88.46	96.93	63.60	74.64
VIM	88.06	97.66	65.60	75.47
Far-OOD				
ID (HMDB51)	Kinetics	UCF	EPIC	HAC
MSP	92.48	87.95	89.07	92.28
Energy	87.81	84.22	92.22	90.23
MaxLogit	90.34	87.91	91.88	91.99
Mahalanobis	95.01	89.34	93.66	94.56
ReAct	97.01	91.45	98.15	95.93
ASH	95.35	92.41	98.46	95.39
GEN	95.45	93.53	99.30	95.66
KNN	96.70	92.33	98.97	97.26
VIM	98.74	94.42	99.63	99.16
Far-OOD				
ID (Kinetics-600)	HMDB	UCF	EPIC	HAC
MSP	71.75	71.49	82.05	75.07
Energy	76.66	72.38	88.05	80.15
MaxLogit	78.43	73.97	84.90	80.30
Mahalanobis	78.84	74.33	82.69	79.51
ReAct	71.88	70.55	84.98	75.15
ASH	80.84	78.20	82.99	85.93
GEN	83.77	84.19	83.30	88.20
KNN	84.30	82.54	83.47	96.58
VIM	81.51	78.38	83.50	99.30

strong and stable performance in model selection. This improvement comes from two factors. First, meta-learning uses knowledge gained from previous tasks to adapt more effectively to new ones, extracting shared patterns and representations that boost generalization. Second, the optimization routines in these methods drive models to efficient, high-quality solutions. By incorporating multimodal embeddings and meta-features, they can map model performance more accurately than simple meta-learners.

3. The poor performance of the no-selection and random-selection baselines highlights the need for OOD model selection. Simply averaging the OOD detection scores of all models yields subpar results, as shown in Fig. 3 and Fig. 2. Some models underperform consistently across datasets, so treating every model equally drags down overall

Model Selection Baselines
No model selection or random selection
MSP (Hendrycks and Gimpel 2017)
Mahalanobis (MD) (Lee et al. 2018a)
Mega Ensemble (ME)
Random Selection (Random)
Simple meta-learners (non-optimization)
Global Best (GB)
ISAC (Kadioglu et al. 2010)
ARGOSMART (AS) (Nikolić, Marić, and Janičić 2013)
Optimization-based meta-learners
ALORS (Misir and Sebag 2017)
NCF (He et al. 2017)
Large language models (LLMs) as model selectors
GPT-4o-mini (OpenAI et al. 2024)

Table 4: Categories of OOD detector selection method baselines in this study.

Ours	Baseline	p-value
M3OOD	ME	0.001
	AS	0.0625
	ISAC	0.0156
	ALORS	0.125
	Random	0.0029
	MSP	0.001
	MD	0.0005
	NCF	0.0039
	GB	0.0312
	GPT-4o mini	0.0015

Table 5: Wilcoxon signed-rank test result (marked bold if no significance). M3OOD is statistically better than all the baselines except AS and ALORS.

effectiveness. While selective ensembles can help (Zhao and Hryniewicki 2019), building and running many models is often too costly. In contrast, M3OOD leverages offline meta-training to learn which single model to choose, avoiding ensemble construction and enabling efficient testing. Moreover, random selection falls short of all meta-learning baselines. This confirms that each meta-learner offers clear gains over random choice, and that picking an OOD detector at random is not advised. In addition, no single OOD detector achieves strong results on every dataset. This is because different OOD detectors target different dataset characteristics, and real-world data vary widely in their properties. Relying on just one method limits the range of solutions and makes it difficult to handle distribution shifts between datasets.

4. LLM as zero-shot model-selector does not perform

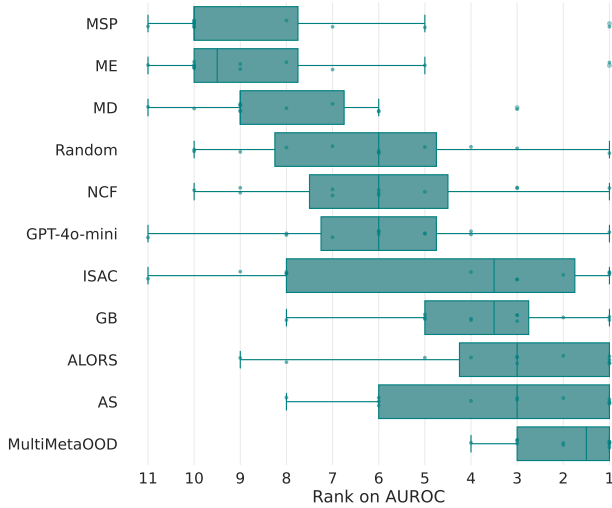


Figure 2: Boxplot of the rank distribution of M3OOD and baselines (the lower, the better). M3OOD is the lowest/best.

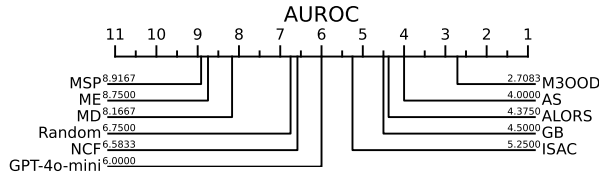


Figure 3: Average rank (lower is better) of methods w.r.t. performance across datasets; M3OOD outperforms all baselines with the lowest rank.

well under multimodal setting. GPT-4o-mini may not be well-suited for capturing the nuanced relationships between multimodal dataset characteristics and OOD detector selection, which likely requires more specialized understanding of how different modalities interact and how various detectors respond to specific types of distribution shifts. This indicates that while LLMs offer accessibility, specialized meta-learning methods still hold substantial advantages for complex, heterogeneous-input settings such as choosing detectors for multimodal OOD detection.

5.1 Ablation Studies and Additional Analysis

Choice of Meta Predictor We evaluate the performances of different choice of meta-predictor f . We test M3OOD with f replaced by a two-layer MLP meta predictor. The result is shown in Fig. 4 left. Similar to what Jiang et al. (2024) observed in their study, XGBoost and other tree-based models produce more reliable and better-performing meta-predictors than neural network alternatives.

Choice of Meta Embedding We compare the performance of M3OOD altered meta-embedding inputs: one that excludes traditional meta-features (Multimodal_emb), and another that excludes multimodal model embeddings (Traditional_mf). Fig. 4 right shows that combining multimodal model embeddings with traditional meta-features leads to

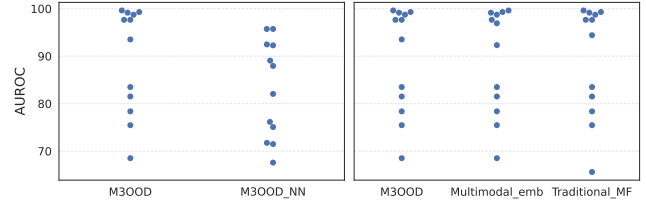


Figure 4: Left: ablation study on different choices of meta-predictor f . Tree-based models have better performance. Right: ablation study on different meta embeddings. M3OOD has better performance over its variants.

improved performance. This suggests that the two types of meta-information provide complementary signals for selecting effective OOD detectors.

Runtime Analysis OOD detection on large datasets is computationally expensive. Fig. 5 compares the runtime of M3OOD components against direct OOD detector execution on the HMDB dataset. While direct OOD detection requires extensive finetuning (HMDB requires 2 mins/ epoch for ~ 40 epochs, while Kinetics needs 10 mins/ epoch for ~ 40 epochs), M3OOD incurs minimal overhead with embedding generation (1577 seconds), meta-learning (57.8 seconds), and online selection (1.6 seconds) for the HMDB dataset. This demonstrates that our model selection approach achieves significant computational savings compared to running OOD detectors directly.

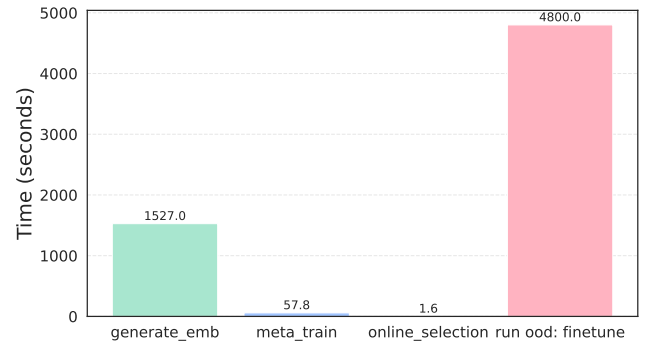


Figure 5: Runtime of M3OOD components vs. time required for multimodal OOD detection on the HMDB dataset. M3OOD incurs a small overhead.

6 Conclusion

In this paper, we propose M3OOD, the *first* framework for multimodal OOD detector selection. Our meta-learner draws on a large set of historical OOD detector and dataset-pair records, using multimodal-related meta-features to guide model choice by learning from prior results. However, M3OOD requires sufficient, high-quality historical dataset pairs, which can limit its performance when such data are scarce or not closely related. For future work, we will expand our evaluation suite to cover a broader range of datasets

and models, thereby improving M3OOD’s meta-learning capabilities. We also plan to integrate an uncertainty estimation component so that M3OOD can return an “I do not know” result when transferable meta-knowledge is insufficient, making it more reliable in challenging scenarios.

References

- Carreira, J.; Noland, E.; Banki-Horvath, A.; Hillier, C.; and Zisserman, A. 2018. A Short Note about Kinetics-600. arXiv:1808.01340.
- Chen, T.; and Guestrin, C. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.
- Cover, T.; and Hart, P. 1967. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1): 21–27.
- Ding, X.; Zhao, Y.; and Akoglu, L. 2024. Fast Unsupervised Deep Outlier Model Selection with Hypernetworks. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM.
- Djurisic, A.; Bozanic, N.; Ashok, A.; and Liu, R. 2023. Extremely Simple Activation Shaping for Out-of-Distribution Detection. arXiv:2209.09858.
- Dong, H.; Nejjar, I.; Sun, H.; Chatzi, E.; and Fink, O. 2023. SimMMDG: A Simple and Effective Framework for Multimodal Domain Generalization. arXiv:2310.19795.
- Dong, H.; Zhao, Y.; Chatzi, E.; and Fink, O. 2024a. MultiOOD: Scaling Out-of-Distribution Detection for Multiple Modalities. *Advances in Neural Information Processing Systems*, 37.
- Dong, H.; Zhao, Y.; Chatzi, E.; and Fink, O. 2024b. MultiOOD: Scaling Out-of-Distribution Detection for Multiple Modalities. arXiv:2405.17419.
- Drori, I.; Liu, L.; Nian, Y.; Koorathota, S. C.; Li, J. S.; Moretti, A. K.; Freire, J.; and Udell, M. 2019. AutoML using Metadata Language Embeddings. arXiv:1910.03698.
- Fang, X.; Xu, W.; Tan, F. A.; Zhang, J.; Hu, Z.; Qi, Y.; Nickleach, S.; Socolinsky, D.; Sengamedu, S.; and Faloutsos, C. 2024. Large Language Models (LLMs) on Tabular Data: Prediction, Generation, and Understanding – A Survey. arXiv:2402.17944.
- Feichtenhofer, C.; Fan, H.; Malik, J.; and He, K. 2019. Slow-Fast Networks for Video Recognition. arXiv:1812.03982.
- He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; and Chua, T.-S. 2017. Neural Collaborative Filtering. In *Proceedings of the 26th International Conference on World Wide Web, WWW ’17*, 173–182. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee. ISBN 9781450349130.
- Hendrycks, D.; Basart, S.; Mazeika, M.; Mostajabi, M.; Steinhardt, J.; and Song, D. X. 2022. Scaling Out-of-Distribution Detection for Real-World Settings. In *ICML*.
- Hendrycks, D.; and Gimpel, K. 2017. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. *Proceedings of International Conference on Learning Representations*.
- Hendrycks, D.; Mazeika, M.; and Dietterich, T. G. 2019. Benchmark for Out-of-Distribution Detection and Robustness. In *International Conference on Learning Representations*.
- Hu, Z.; Song, L.; Zhang, J.; Xiao, Z.; Wang, J.; Chen, Z.; and Xiong, H. 2024. Rethinking llm-based preference evaluation. *arXiv e-prints*, arXiv–2407.
- Jiang, M.; Hou, C.; Zheng, A.; Han, S.; Huang, H.; Wen, Q.; Hu, X.; and Zhao, Y. 2024. ADGym: Design Choices for Deep Anomaly Detection. *Advances in Neural Information Processing Systems*, 36.
- Jomaa, H. S.; Schmidt-Thieme, L.; and Grabocka, J. 2021. Dataset2vec: Learning dataset meta-features. *Data Mining and Knowledge Discovery*, 35(3): 964–985.
- Kadioglu, S.; Malitsky, Y.; Sellmann, M.; and Tierney, K. 2010. ISAC - Instance-Specific Algorithm Configuration. In *ECAI*, volume 215 of *Frontiers in Artificial Intelligence and Applications*, 751–756. IOS Press.
- Kay, W.; Carreira, J.; Simonyan, K.; Zhang, B.; Hillier, C.; Vijayanarasimhan, S.; Viola, F.; Green, T.; Back, T.; Natsev, P.; Suleyman, M.; and Zisserman, A. 2017. The Kinetics Human Action Video Dataset. arXiv:1705.06950.
- Kuehne, H.; Jhuang, H.; Garrote, E.; Poggio, T.; and Serre, T. 2011. HMDB: A large video database for human motion recognition. In *2011 International Conference on Computer Vision*, 2556–2563.
- Lee, K.; Lee, K.; Lee, H.; and Shin, J. 2018a. A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. In *Advances in Neural Information Processing Systems*, 7167–7177.
- Lee, K.; Lee, K.; Lee, H.; and Shin, J. 2018b. A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. In Bengio, S.; Wallach, H.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.
- Li, L.; Ji, W.; Wu, Y.; Li, M.; Qin, Y.; Wei, L.; and Zimmermann, R. 2024a. Panoptic Scene Graph Generation with Semantics-Prototype Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(4): 3145–3153.
- Li, L.; Qin, Y.; Ji, W.; Zhou, Y.; and Zimmermann, R. 2024b. Domain-wise invariant learning for panoptic scene graph generation. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3165–3169. IEEE.
- Li, L.; Wang, C.; Qin, Y.; Ji, W.; and Liang, R. 2023. Biased-predicate annotation identification via unbiased visual predicate representation. In *Proceedings of the 31st ACM International Conference on Multimedia*, 4410–4420.
- Li, S.; Cai, P.; Zhou, Y.; Ni, Z.; Liang, R.; Qin, Y.; Nian, Y.; Tu, Z.; Hu, X.; and Zhao, Y. 2025a. Secure on-device video ood detection without backpropagation. *arXiv preprint arXiv:2503.06166*.
- Li, S.; Gong, H.; Dong, H.; Yang, T.; Tu, Z.; and Zhao, Y. 2025b. Dpu: Dynamic prototype updating for multimodal

- out-of-distribution detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 10193–10202.
- Li, S.; Qu, J.; Zhou, Y.; Qin, Y.; Yang, T.; and Zhao, Y. 2025c. Treble counterfactual vlms: A causal approach to hallucination. *arXiv preprint arXiv:2503.06169*.
- Liang, S.; Li, Y.; and Srikant, R. 2017. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. *arXiv: Learning*.
- Liang, S.; Li, Y.; and Srikant, R. 2018. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*.
- Liu, W.; Wang, X.; Owens, J.; and Li, Y. 2020a. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems*, volume 33, 21464–21475.
- Liu, W.; Wang, X.; Owens, J.; and Li, Y. 2020b. Energy-based Out-of-distribution Detection. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 21464–21475. Curran Associates, Inc.
- Liu, X.; Lochman, Y.; and Zach, C. 2023. GEN: Pushing the Limits of Softmax-Based Out-of-Distribution Detection. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 23946–23955.
- Ming, Y.; Cai, Z.; Gu, J.; Sun, Y.; Li, W.; and Li, Y. 2022. Delving into Out-of-Distribution Detection with Vision-Language Representations. *arXiv:2211.13445*.
- Misir, M.; and Sebag, M. 2017. Alors: An algorithm recommender system. *Artif. Intell.*, 244: 291–314.
- Munro, J.; and Damen, D. 2020. Multi-Modal Domain Adaptation for Fine-Grained Action Recognition. *arXiv:2001.09691*.
- Nikolić, M.; Marić, F.; and Janičić, P. 2013. Simple algorithm portfolio for SAT. *Artificial Intelligence Review*, 40(4): 457–465.
- OpenAI; Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; et al. 2024. GPT-4 Technical Report. *arXiv:2303.08774*.
- Park, N.; Rossi, R. A.; Ahmed, N.; and Faloutsos, C. 2023. MetaGL: Evaluation-Free Selection of Graph Learning Models via Meta-Learning. In *The Eleventh International Conference on Learning Representations*.
- Qin, Y.; Zhang, Y.; Nian, Y.; Ding, X.; and Zhao, Y. 2025. MetaOOD: Automatic Selection of OOD Detection Models. *arXiv:2410.03074*.
- Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. *arXiv:2103.00020*.
- Shi, T.; Wu, Y.; Song, L.; Zhou, T.; and Zhao, J. 2025. Efficient reinforcement finetuning via adaptive curriculum learning. *arXiv preprint arXiv:2504.05520*.
- Song, L.; Cui, Y.; Luo, A.; Lecue, F.; and Li, I. 2024. Better Explain Transformers by Illuminating Important Information. *EACL 2024 (Findings)*.
- Song, L.; Dai, Y.; Prabhu, V.; Zhang, J.; Shi, T.; Li, L.; Li, J.; Savarese, S.; Chen, Z.; Zhao, J.; et al. 2025a. CoAct-1: Computer-using Agents with Coding as Actions. *arXiv preprint arXiv:2508.03923*.
- Song, L.; Ding, X.; Zhang, J.; Shi, T.; Shimizu, R.; Gupta, R.; Liu, Y.; Kang, J.; and Zhao, J. 2025b. Discovering knowledge deficiencies of language models on massive knowledge base. *COLM 2025*.
- Song, L.; Zhang, J.; Cheng, L.; Zhou, P.; Zhou, T.; and Li, I. 2023. Nlpbench: Evaluating large language models on solving nlp problems. *arXiv preprint arXiv:2309.15630*.
- Song, L.; Zhang, J.; Yang, T.; and Goto, M. 2022. Adaptive Ranking-based Sample Selection for Weakly supervised Class-imbalanced Text Classification. *EMNLP 2022 (Findings)*.
- Soomro, K.; Zamir, A. R.; and Shah, M. 2012. UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. *arXiv:1212.0402*.
- Sun, Y.; Cheng, D.; Yang, T.; Ji, Y.; Zhang, S.; Zhu, M.; Xiong, X.; Fan, Q.; Liang, M.; Pei, D.; et al. 2023. Efficient and robust KPI outlier detection for large-scale datacenters. *IEEE Transactions on Computers*, 72(10): 2858–2871.
- Sun, Y.; Guo, C.; and Li, Y. 2021. ReAct: Out-of-distribution Detection With Rectified Activations. *arXiv:2111.12797*.
- Ulmer, D.; Meijerink, L.; and Cinà, G. 2020. Trust issues: Uncertainty estimation does not enable reliable ood detection on medical tabular data. In *Machine Learning for Health*, 341–354. PMLR.
- Vanschoren, J. 2018. Meta-learning: A survey. *arXiv preprint arXiv:1810.03548*.
- Wang, H.; Li, Y.; Yao, H.; and Li, X. 2023. CLIPN for Zero-Shot OOD Detection: Teaching CLIP to Say No. *arXiv:2308.12213*.
- Wang, H.; Li, Z.; Feng, L.; and Zhang, W. 2022. ViM: Out-Of-Distribution with Virtual-logit Matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Wang, H.; and Schmid, C. 2013. Action Recognition with Improved Trajectories. In *2013 IEEE International Conference on Computer Vision*, 3551–3558.
- Wang*, S.; Song*, L.; Zhang, J.; Shimizu, R.; Luo, A.; Yao, L.; Chen, C.; McAuley, J.; and Wu, H. 2024. Template Matters: Understanding the Role of Instruction Templates in Multimodal Language Model Evaluation and Training. *arXiv preprint arXiv:2412.08307*.
- Wolpert, D. H.; and Macready, W. G. 1997. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1): 67–82.
- Xu, H.; Liu, K.; Yao, Z.; Yu, P. S.; Li, M.; Ding, K.; and Zhao, Y. 2025a. LEGO-Learn: Label-Efficient Graph Open-Set Learning. *arXiv:2410.16386*.
- Xu, H.; Yao, Z.; Dong, Y.; Wang, Z.; Rossi, R. A.; Li, M.; and Zhao, Y. 2025b. Few-Shot Graph Out-of-Distribution Detection with LLMs. *arXiv:2503.22097*.

- Xu, H.; Yao, Z.; Zhang, X.; Wang, Z.; He, L.; Dong, Y.; Yu, P. S.; Li, M.; and Zhao, Y. 2025c. GLIP-OOD: Zero-Shot Graph OOD Detection with Graph Foundation Model. *arXiv:2504.21186*.
- Xu, L.; Hutter, F.; Shen, J.; Hoos, H. H.; and Leyton-Brown, K. 2012. SATzilla2012: Improved algorithm selection based on cost-sensitive classification models. *Proceedings of SAT Challenge*, 57–58.
- Xue, F.; He, Z.; Zhang, Y.; Xie, C.; Li, Z.; and Tan, F. 2024. Enhancing the Power of OOD Detection via Sample-Aware Model Selection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17148–17157.
- Yang, J.; Zhou, K.; Li, Y.; and Liu, Z. 2021. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*.
- Yi, C.; Yang, S.; Wang, Y.; Li, H.; Tan, Y.-P.; and Kot, A. C. 2023. Temporal Coherent Test-Time Optimization for Robust Video Classification. *arXiv:2302.14309*.
- Zhang, A.; Fei, H.; Yao, Y.; Ji, W.; Li, L.; Liu, Z.; and Chua, T.-S. 2023. VPGTrans: Transfer Visual Prompt Generator across LLMs. *arXiv:2305.01278*.
- Zhao, Y.; and Hryniewicki, M. K. 2019. DCSO: dynamic combination of detector scores for outlier ensembles. *arXiv preprint arXiv:1911.10418*.
- Zhao, Y.; Rossi, R.; and Akoglu, L. 2021. Automatic Unsupervised Outlier Model Selection. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 4489–4502. Curran Associates, Inc.
- Zhao, Y.; Zhang, S.; and Akoglu, L. 2022. Toward Unsupervised Outlier Model Selection. In *2022 IEEE International Conference on Data Mining (ICDM)*, 773–782. Los Alamitos, CA, USA: IEEE Computer Society.

A Pseudo-code for Meta-train and Online Model Selection

We discussed meta-training and online model selection in Section §3.3, 3.4 and § 3.5, respectively. Below are the pseudo-code for the two phases.

Algorithm 1: Offline OOD detection meta-learner training

Input: Meta-train database $\mathcal{D}_{\text{train}}$ composed of K -modality data, model set $\mathcal{M}$
Output: Meta-learner f for OOD detection model selection
 Train and evaluate $\mathcal{M}$ on $\mathcal{D}_{\text{train}}$ to get performance matrix $\mathbf{P}$
for $i \in \{1, \dots, n\}$ **do**
 Extract data embedding $E_i^{\text{meta}} = \psi(D_i) = [\psi(x_1), \dots, \psi(x_K)]$
 for $j \in \{1, \dots, m\}$ **do**
 Encode methods set as $E_j^{\text{model}} = \phi(\mathcal{M}, M_j)$
 Train f by Eq. (1) with the j -th model on the i -th dataset
 end for
end for
return the meta-learner f

Algorithm 2: Online OOD detection model selection

Input: the meta-learner f , New ID-OOD dataset pair D_{new}
Output: Selected model for D_{new}
 Extract data embedding, $E_{\text{test}}^{\text{data}} := \psi(D_{\text{new}})$
for $j \in \{1, \dots, m\}$ (for clarity, written as a for loop) **do**
 Encode methods set as $E_j^{\text{model}} = \phi(\mathcal{M}, M_j)$
 Predict the j -th model performance by the meta-learner f ,
 i.e., $\hat{\mathbf{P}}_{\text{new},j} := f(E_{\text{new}}^{\text{data}}, E_j^{\text{model}})$
end for
return the model with the highest predicted perf. by Eq. (2)

B Details of Meta-features

We introduced the multimodal model embeddings and meta-features for capturing multimodal OOD data characteristics in Section §3.4. Table 6 lists the complete set of meta-features we construct. Part of the meta-features are based on (Yi et al. 2023; Vanschoren 2018; Wang and Schmid 2013)

C Details of Baselines

Section §4 introduces the model selection baselines, which we choose based on prior work in meta-learning for unsupervised model selection (Zhao, Rossi, and Akoglu 2021; Zhao, Zhang, and Akoglu 2022; Jiang et al. 2024; Park et al. 2023). These baselines are grouped into four categories, as shown in main Tab. 4. Further details are provided below:

(a) **No model selection or random selection:** always employs either the ensemble of all the models or the same single model, or randomly selects a model: (1) **Maximum Soft-max Probability (MSP)** (Hendrycks and Gimpel 2017), a popular OOD detector that uses the maximum softmax score of a neural network’s logits as threshold to identify whether an input belongs to the distribution the network was trained on. (2) **Mahalanobis (MD)** (Lee et al. 2018a) computes the distance between a sample’s features and class means using the Mahalanobis metric, treating lower distances as more

Meta-feature	Definition
<i>Video and Optical Flow Related Meta-features</i>	
Clip length T	Number of RGB frames
RGB height H	Height of each RGB frame in pixels
RGB width W	Width of each RGB frame in pixels
RGB aspect ratio	H/W
Flow height H'	Height of each optical-flow frame
Flow width W'	Width of each optical-flow frame
Flow aspect ratio	H'/W'
Colourfulness index	Hasler–Süsstrunk measure computed from all pixels
Edge density	Fraction of Canny edge pixels, averaged over time
GLCM entropy	Average grey-level co-occurrence entropy over frames
HoF histogram	Eight-bin, magnitude-weighted histogram of flow orientations covering $(-\pi, \pi]$
<i>Basic Statistics</i>	
μ_I (Clip mean)	Mean value
σ_I (Clip std)	Standard deviation value
skew_I	Skewness of distribution
kurt_I	Kurtosis of distribution
$\min_I$	Minimum value
$\max_I$	Maximum value
med_I	Median value
IQR_I	Interquartile range of intensities
Gini_I	Gini coefficient of values
MAD_I	Median absolute deviation value
AAD_I	Mean absolute deviation value
CV_I	Coefficient of variation (std/mean)
$p_{\text{out},I}^{1\%}$	Proportion outside 1st–99th percentile
$p_{\text{out},I}^{3\sigma}$	Proportion outside $\mu_I \pm 3\sigma_I$
μ_M (Flow mean)	Mean of optical-flow magnitudes
σ_M (Flow std)	Standard deviation of optical-flow magnitudes
IQR_M	Interquartile range of flow magnitudes
$p_{\text{out},M}^{1\%}$	Proportion outside 1st–99th percentile of flow magnitudes
$p_{\text{out},M}^{3\sigma}$	Proportion outside $\mu_M \pm 3\sigma_M$ of flow magnitudes

Table 6: Details of the meta-features. Meta-features include CLIP-based per-frame meta-features and the grayscale and motion meta-features.

likely to be in-distribution. (3) **Mega Ensemble (ME)** averages OOD scores from all the models for a given dataset without performing model selection but rather using *all* the models. (4) **Random Selection (Random)** selects a model at

random from the set of available candidate detectors.

(b) **Simple meta-learners** that do not involve optimization: (5) *Global Best (GB)* is the *simplest meta-learner* that selects the model with the largest average performance across all meta-train datasets. GB does *not* use any meta-features. (6) *ISAC* (Kadioglu et al. 2010) clusters the meta-train datasets based on meta-features. Given a new dataset pair, it identifies its closest cluster and selects the best-performing model in that cluster. (7) *ARGOSMART (AS)* (Nikolić, Marić, and Janičić 2013) finds the closest meta-train dataset (1 nearest neighbor) to a given test dataset in terms of meta-feature distance, and selects the model with the best performance on the INN dataset.

(c) **Optimization-based meta-learners** which involves a learning process: (8) *ALORS* (Misir and Sebag 2017) factorizes the performance matrix to extract latent factors and estimate the performance matrix as the dot product of the latent factors. A regressor maps meta-features to these latent factors. (9) *NCF* (He et al. 2017) replaces the dot product used in ALORS with a more general neural architecture that predicts performance by combining the linearity of matrix factorization and non-linearity of deep neural networks.

(d) **Large language models (LLMs) as a model selector**: (10) *GPT-4o mini* (OpenAI et al. 2024) used as zero-shot meta-selector. The dataset and method descriptions are directly provided to the LLM, allowing it to select the methods based on these descriptions. Note there is no meta-learning here. Details are in Supplementary Material §3.

C.1 Random Selection

The Random Selection baseline uses random seed 42.

C.2 GPT-4o-mini

GPT-4o-mini (OpenAI et al. 2024) is used as one of the baselines, serving as a zero-shot meta-selector. The text inputs are as follows:

Datasets Descriptions *EPIC-Kitchens*: A large-scale egocentric dataset collected by 32 participants recording daily kitchen activities in their homes. We use the subset from the Multimodal Domain Adaptation benchmark, comprising 4,871 clips from the 8 largest action classes in sequence P22: put, take, open, close, wash, cut, mix, and pour. Modalities include video, optical flow, and audio.

HAC: Contains 3,381 video clips of 7 actions: sleeping, watching TV, eating, drinking, swimming, running, and opening door, performed by humans, animals, and cartoon figures. Includes video, optical flow, and audio modalities.

UCF101: Comprises 13,320 YouTube video clips covering 101 actions with significant diversity in motion, appearance, and background. Modalities include video and optical flow.

Kinetics-600: A large-scale dataset of 480k 10-second clips spanning 600 actions. We use a subset of 229 classes (57,205 clips) to reduce class overlap with other datasets. Optical flow is extracted at 24 FPS using the TV-L1 algorithm, totaling 114,410 samples. Final modalities include video, audio, and optical flow.

OOD Detector Descriptions **MSP**: Implements the Maximum Softmax Probability (MSP) Thresholding baseline for OOD detection.

EnergyBased: Calculates the negative energy for a vector of logits. This value is used as the outlier score.

MaxLogit: Implements the Max Logit Method for OOD Detection, as proposed in Scaling Out-of-Distribution Detection for Real-World Settings.

Mahalanobis: Calculates a class center for each class and a shared covariance matrix from the data. It also uses ODIN preprocessing.

ReAct: Clips the activations in some layer of the network (backbone) and forward propagates the result through the remainder of the model (head). In the paper, ReAct is applied to the penultimate layer of the network. The output of the network is then passed to an outlier detector that maps the output of the model to outlier scores.

ASH: Prunes the activations in some layer of the network (backbone) by removing a certain percentile of the highest activations. The remaining activations are modified, depending on the particular variant selected, and propagated through the remainder (head) of the network. Then uses the energy-based outlier score. This approach has been shown to increase OOD detection rates while maintaining ID accuracy.

GEN: Utilizes the entropy of the softmax output as a measure of confidence. In-distribution samples are expected to have higher confidence (lower entropy), while OOD samples will exhibit lower confidence (higher entropy).

ViM: Implements Virtual Logit Matching (ViM) from the paper ViM: Out-Of-Distribution Detection with Virtual-logit Matching.

KNN: Implements the detector from the paper Out-of-Distribution Detection with Deep Nearest Neighbors. Fits a nearest neighbor model to the IN samples and uses the distance from the nearest neighbor as the outlier score.

Prompt The prompt provided to the LLM is structured as follows, with text descriptions of both the datasets and models provided. To ensure consistency, we set *temperature* parameter to 0, and *top-p* parameter to 0.999.

[Dataset descriptions provided]

Your task is to select the best OOD detection method for a set of 24 test ID-OOD dataset pairs. You will be provided with descriptions of both the ID-OOD dataset pairs and the available OOD detection methods. You should pick the best model that has the highest AUROC metric. For each dataset pair, output the recommended OOD detection method in the format: 'Recommended Method: [Recommended Method]'.

[Model descriptions provided]

D Near-OOD and Far-OOD Setup

M3OOD utilizes five video datasets comprising more than 85,000 video clips in total. These datasets differ in the number of classes, which range from 7 to 229, and in size, rang-

ing from 3,000 to 57,000 clips. Video and optical flow are used as different types of modalities. Details of the five datasets are in § 3.1.1.

In the **Near-OOD** setup, four datasets are used. EPIC-Kitchens 4/4 is derived from the EPIC-Kitchens Domain Adaptation dataset (Munro and Damen 2020), with four classes used for training as ID and four for testing as OOD, totaling 4,871 video clips. HMDB51 25/26 and UCF101 50/51 are constructed from HMDB51 (Kuehne et al. 2011) and UCF101 (Soomro, Zamir, and Shah 2012), with 6,766 and 13,320 clips, respectively. Kinetics-600 129/100 uses 229 classes selected from Kinetics-600 (Carreira et al. 2018), each with about 250 clips (57,205 total); 129 classes are used as ID and 100 as OOD.

In the **Far-OOD** setup, HMDB51 and Kinetics-600 are used as ID datasets. For HMDB51 as ID, OOD datasets include UCF101, EPIC-Kitchens, HAC, and Kinetics-600. To avoid class overlap, we exclude 8 overlapping classes from HMDB51 (leaving 43 ID classes) and remove 31 overlapping classes from UCF101 (resulting in 70 OOD classes). For the remaining datasets, no overlap exists and their original classes are used. For Kinetics-600 as ID, the same OOD datasets are adopted. We use the same 229-class subset from the Near-OOD setup to reduce overlap. For UCF101, 11 overlapping classes are removed, leaving 90 classes as OOD. Other datasets are used as-is due to no class overlap.

E Additional Experiment Setting

We select the parameters for M3OOD and M3OOD_NN (used in the ablation study) through grid search. The final parameter configurations are provided in the code repo.

F Additional Results

Figure 6 shows the dataset embeddings visualization, with the embeddings reduced to 2D using t-SNE. We observe clear clustering patterns that reflect underlying similarities across datasets. For instance, datasets originating from the same source or sharing overlapping label spaces, such as Kinetics-HMDB and Kinetics-UCF, are located closely, indicating that the meta-features capture alignment in distribution or content. Similarly, HMDB-EPIC and HMDB-Kinetics are proximal to HMDB, suggesting consistency in the extracted features when paired with other datasets. Moreover, datasets involving HAC (e.g., Kinetics-HAC, HMDB-HAC) appear in a distinct region, separated from others. This spatial distinction implies that the HAC dataset exhibits different properties—such as lower visual diversity, temporal resolution, or action granularity—compared to datasets like Kinetics and EPIC. This separation also highlights the ability of the meta-features to reflect meaningful dataset differences relevant for model selection and generalization.

G Details on Notations

The following notations are used in Main Fig. 1 and Main Alg. 2 for comprehensive M3OOD overview.

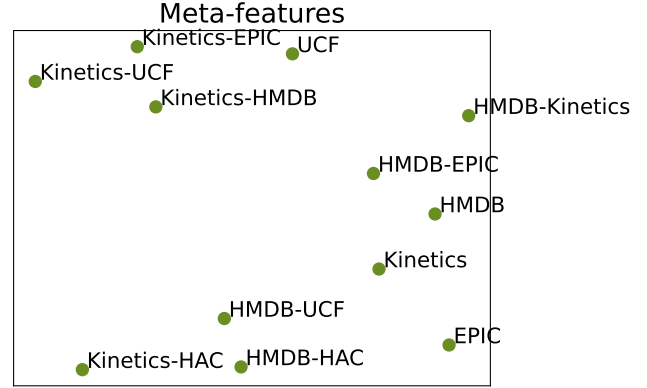


Figure 6: Visualization of dataset embeddings.

Notations	Description
$\mathcal{L}$	Training Loss
$\mathcal{M}$	# OOD Detection Methods
$\mathcal{N}$	# Dataset Pairs
D	Dataset Pair
ϕ	Embedding Notation for OOD Detectors
ψ	Embedding Notation for Dataset Pairs
E	Embeddings for datasets and models
$P_{i,j}$	Performance of OOD Detector j on Dataset Pair i
$\hat{P}_{i,j}$	Predicted Performance of OOD Detector j on Dataset Pair i
f	Meta-predictor

Table 7: Notations with details used in Main Fig. 1 and Alg. 2.