MASKED DIFFUSION MODELING FOR ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Anomaly detection aims to identify samples that deviate from the nominal data distribution and is central to many safety-critical applications. However, developing effective anomaly detection methods for categorical and discrete data remains challenging and relatively underexplored. Masked diffusion models provide a natural way to model such data by learning to recover masked values from remaining visible context. In this paper, we propose *Masked Diffusion for Anomaly Detection* (MaskDiff-AD), a forward-only method based on masked diffusion models trained only on nominal data. Given a test sample, MaskDiff-AD calculates anomaly scores from the difficulty of reconstructing randomly masked coordinates, yielding a content-sensitive score that operates directly on discrete state spaces while avoiding reverse-time sampling. We also develop a non-parametric variant of MaskDiff-AD and provide theoretical analysis by characterizing Type-I and Type-II errors under a fixed detection threshold. Experiments on fourteen tabular datasets show that MaskDiff-AD achieves the best overall average rank among all evaluated methods, and experiments on four text anomaly detection datasets also demonstrate competitive performance against the baselines.

1 INTRODUCTION

Anomaly detection aims to identify observations that deviate significantly from the majority of the data (i.e., the data-generating distribution) (Chandola et al., 2009; Ruff et al., 2021; Pang et al., 2021). As a core problem in machine learning and statistics, it has been studied for decades and plays an important role in a wide range of applications, including healthcare (Salem et al., 2013; Pachauri & Sharma, 2015), finance (Ahmed et al., 2016), cybersecurity (Ahmed et al., 2016), manufacturing (Susto et al., 2017), particle physics (Fraser et al., 2022), and geospatial analysis (Yairi et al., 2006). Despite this long history, the increasing prevalence of large-scale datasets with discrete or mixed data types has exposed important limitations of many classical methods (Taha & Hadi, 2019). Many modern anomaly detection problems involve categorical, discrete, or mixed-type data, including tabular records with categorical attributes, transaction and claims data with heterogeneous feature types, and tokenized text sequences (Han et al., 2022; Li et al., 2025). These data types are not naturally suited to methods built around Euclidean geometry or continuous densities. These challenges motivate the development of anomaly detection methods that are well-suited to the structural characteristics of discrete type data, while being scalable, interpretable, and sufficiently expressive.

Recently, diffusion models provide a flexible framework for learning complex data distributions (Sohl-Dickstein et al., 2015; Ho et al., 2020). Among them, discrete diffusion models, especially masked diffusion models, have further extended this capability to discrete domains such as text, music, and molecular data (Sahoo et al., 2024; Austin et al., 2021). Notably, on certain tasks, masked diffusion models have demonstrated superior performance compared to autoregressive approaches (Zheng et al., 2025; Prabhudesai et al., 2025). However, as with other generative models, the majority of research on diffusion models has focused on improving sampling efficiency and enabling controllable generation (e.g., Zheng et al. (2025); Cardei et al. (2025); Uehara et al. (2024)). In contrast, relatively few studies have explored their use, particularly that of masked diffusion models, for anomaly detection.

In the literature, one notable work has explored the use of (continuous) diffusion models for anomaly detection: the Diffusion Time Estimation (DTE) framework (Livernoche et al., 2024). This approach estimates the posterior distribution over the diffusion time and uses the inferred corruption level as

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

an anomaly score, thereby avoiding the need for expensive reverse-time sampling. However, a key component of this framework relies on the Gaussian corruption structure, with a possible discrete analogue in uniform discrete diffusion models. As we show in Section 3, this approach does not readily extend to masked diffusion models. In particular, due to the absorbing masking mechanism, the DTE-style posterior becomes ill-posed: it either collapses to identical values across all clean categorical samples or depends solely on the masking pattern of the corrupted observation, rather than the underlying distribution of the test data.

Masked diffusion models (MDM) have recently gained increasing attention for modeling discrete data (Sahoo et al., 2024). They learn to reconstruct clean categorical values from partially masked observations, such as learning to predict masked tokens from the remaining context. Motivated by the masking procedure, we propose *Masked Diffusion for Anomaly Detection* (MaskDiff-AD). The key idea is to employ a *mask-corruption level* tailored to the masked discrete diffusion process, and to use the corresponding *single-step reconstruction difficulty* as the anomaly score. In other words, instead of asking at which corruption time a test sample becomes indistinguishable from noisy normal data, MaskDiff-AD asks how difficult it is to reconstruct masked coordinates from the remaining visible context under a model trained only on normal data. A normal sample should have coordinates that are mutually consistent and therefore easy to predict from one another, whereas an anomalous sample should induce larger masked reconstruction surprisal. This yields a content-sensitive anomaly score without reverse-time sampling or iterative imputation.

The contributions of our work are summarized as follows:

- We formulate anomaly detection on discrete state spaces using masked diffusion models and show why a direct DTE-style time posterior degenerates in this setting.
- We introduce MaskDiff-AD, a masked reconstruction-based anomaly detection method that is trained *only on the nominal samples*, computationally efficient, and does not require reverse-time sampling. We also develop both non-parametric and parametric implementations of MaskDiff-AD.
- We provide theoretical analysis for the proposed reconstruction score, including guarantees on Type-I and Type-II errors.
- We demonstrate the effectiveness of our MaskDiff-AD with extensive experiments on categorical tabular, mixed-type tabular, and text anomaly detection benchmarks.

Related Work. Our work is connected to several lines of research. Classical anomaly detection methods include distance-based approaches such as k-nearest neighbors (Ramaswamy et al., 2000), tree-based methods such as Isolation Forest (Liu et al., 2008), and distributional or copula-based scores such as ECOD (Li et al., 2023) and COPOD (Li et al., 2020). These methods are often efficient and broadly applicable, but their performance can depend strongly on the chosen representation or distance, especially for categorical data (Taha & Hadi, 2019; Chawda et al., 2022). Recent tabular anomaly detection methods seek to capture richer feature dependencies, including masked cell modeling (MCM) (Yin et al., 2024), non-parametric transformers for anomaly detection (NPT-AD) (Thimonier et al., 2024), internal contrastive learning (ICL) (Shenkar & Wolf, 2022), CompreX (Akoglu et al., 2012), and decomposed representation learning (Ye et al., 2025). MCM is particularly related to our work because both exploit masked feature dependencies for anomaly detection. MCM is formulated for numerical tabular data, learning input-dependent masks and scoring Euclidean reconstruction error in a continuous representation; our MaskDiff-AD instead operates directly on discrete states, using masks from a prescribed absorbing diffusion process and scoring masked conditional surprisal.

Generative models for anomaly detection have been extensively studied, especially for image data. Representative examples include autoencoder and variational-autoencoder based detectors (Sakurada & Yairi, 2014; An & Cho, 2015), hybrid autoencoding-density models such as DAGMM (Zong et al., 2018), GAN-based detectors such as AnoGAN and GANomaly (Schlegl et al., 2017; Akcay et al., 2018), flow-based visual detectors such as DifferNet (Rudolph et al., 2021), and LLM-based detectors such as AnoLLM (Tsai et al., 2025). Diffusion models have also recently been explored for anomaly detection through reconstruction-based, density-based, and hybrid scoring mechanisms (Liu et al., 2025). Representative diffusion-based anomaly detection methods include medical and visual reconstruction methods such as AnoDDPM (Wyatt et al., 2022), DDAD (Mousakhan et al., 2024), and DiffusionAD (Zhang et al., 2025). These methods are primarily developed for continuous data domains, especially for (medical) image data, where anomaly scores are often derived from denoising error, likelihood surrogates, or reconstruction discrepancy after reverse diffusion.

Diffusion time estimation (DTE) (Livernoche et al., 2024) is a recent method for anomaly detection and is efficient because it replaces expensive reverse-chain sampling with a single posterior evaluation, yet retains strong empirical performance. However, its construction relies on the geometry of Gaussian corruption and does not remain discriminative under absorbing masked corruption. Our work instead adapts the diffusion perspective to discrete data by replacing time-posterior estimation with masked reconstruction difficulty. This keeps the computational advantage of forward-only scoring while producing a content-sensitive anomaly signal on the original discrete state space.

2 PRELIMINARIES

We consider anomaly detection over a general discrete state space $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d$ and denote each data point as a vector $\mathbf{x} = (x_1, \ldots, x_d) \in \mathcal{X}$, where each coordinate $x_j \in \mathcal{X}_j$ takes values in a finite (and potentially different) state space. The goal is to distinguish samples generated from a nominal data-generating distribution from anomalous samples that deviate from this distribution. We denote the nominal distribution by q_0 and assume that the training data consist *only* of normal samples $\mathcal{D}_{\text{tr}} = \{\mathbf{x}_i\}_{i=1}^n$ that are independently and identically sampled from q_0. At test time, a sample may either follow the same nominal distribution or arise from an unknown anomalous distribution q' that differs significantly from q_0. The anomalous distribution is not specified during training and no labeled anomalous examples are assumed to be available for training.

Given a test sample $\mathbf{x} \in \mathcal{X}$, an anomaly score is assigned such that a larger score indicates that $\mathbf{x}$ is less likely to have been generated from the normal regime. Formally, we define an anomaly detector as a scoring function $S : \mathcal{X} \to \mathbb{R}$ and declare a test sample $\mathbf{x}$ to be anomalous if $S(\mathbf{x}) > \gamma$, where γ is the detection threshold chosen by controlling the false alarm rate. One example is to choose γ by requiring $\mathbb{P}_{\mathbf{x}\sim q_0}\big(S(\mathbf{x}) > \gamma\big) \le \alpha$ for a prescribed significance level α. Since the anomalous distribution is unknown, the main challenge is to construct such a score function that is sensitive to distributional deviations while relying only on normal training data.

2.1 MASKED DIFFUSION MODELS

Masked diffusion models (MDM) provide a generative framework for modeling discrete data by learning to reconstruct clean samples from randomly masked observations at each generation step (Sahoo et al., 2024). Trained on nominal data, MDMs can be viewed as learning the underlying distributional characteristics of the nominal distribution q_0. As the name suggests, MDMs introduce a special mask state m that does not belong to any $\mathcal{X}_j$, resulting in the augmented space:

$$\widetilde{\mathcal{X}} := \widetilde{\mathcal{X}}_1 \times \cdots \times \widetilde{\mathcal{X}}_d, \quad \text{with } \widetilde{\mathcal{X}}_j := \mathcal{X}_j \cup \{\mathsf{m}\}.$$

MDMs include a forward and a reverse process. Given a clean nominal sample $\mathbf{x}_0 \sim q_0$, the forward process constructs a family of intermediate random variables $\{\mathbf{x}_t\}_{t\in[0,1]}$ that interpolate between clean data $\mathbf{x}_0$ and the fully masked state $\mathbf{x}_1 = \{\mathsf{m}, \ldots, \mathsf{m}\}$. This is implemented as a *masking (a.k.a., absorbing-state) process* with increasing probability to mask over time. Specifically, let $\alpha_t \in [0, 1]$ be a monotone decreasing function with $\alpha_0 \approx 1$ and $\alpha_1 = 0$. With the standard setup in Sahoo et al. (2024); Shi et al. (2024), one can show that the conditional forward distribution is

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \prod_{j=1}^{d} \mathrm{Cat}\big(x_{t,j};\, \alpha_t \delta_{x_{0,j}} + (1 - \alpha_t)\delta_{\mathsf{m}}\big), \tag{1}$$

where $\delta_{x_{0,j}}$ is a delta function centered at $x_{0,j}$. Moreover, once a coordinate becomes masked, it remains masked for all later times, reflecting the absorbing property.

Then, one way to define the reverse process is to model $q_{s|t,0}(\mathbf{x}_s \mid \mathbf{x}_t, \mu_\theta(\mathbf{x}_t, t))$ for $s < t$. Here $\mu_\theta^j(\mathbf{x}_t, t) := p_\theta^j(\cdot \mid \mathbf{x}_t)$ is a distribution over $\mathcal{X}_j$ and is intended to approximate the conditional law of the clean coordinate $x_{0,j}$ given the masked view $\mathbf{x}_t$. To train for p_θ^j, one can employ the standard masked diffusion training objective, which is a weighted masked reconstruction loss, as follows:

$$\mathcal{L}_{\text{MDM}}(\theta) = \int_0^1 \mathbb{E}_{\mathbf{x}_0\sim q_0,\, \mathbf{x}_t\sim q_{t|0}(\cdot|\mathbf{x}_0)} \Big(\sum_{j:x_{t,j}=\mathsf{m}} w(t) \log p_\theta^j(x_{0,j} \mid \mathbf{x}_t) \Big) \mathrm{d}t, \tag{2}$$

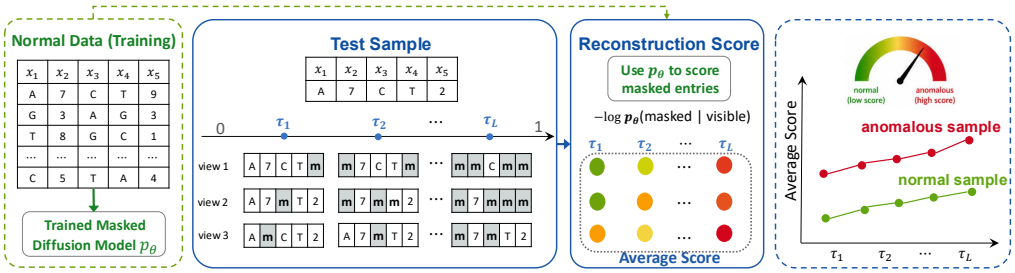


Figure 1: Overview of MaskDiff-AD. We first generate masked probe views of a test sample at multiple levels $\tau_1, \ldots, \tau_L$, and score each view by the reconstruction surprisal of the masked entries using a trained MDM p_θ on normal-only data. The final anomaly score averages these scores across views and mask levels; larger score values indicate stronger anomaly evidence.

where $w(t) := \alpha_t'/(1-\alpha_t)$ is a function of α_t. Essentially, the objective learns to reconstruct the clean data from the partially masked state. After training on nominal data, the learned $p_\theta^j(\cdot \mid \mathbf{x}_t)$ captures the data distribution of the normal regime. Intuitively, a test sample should be considered normal if its coordinates are predictable from the rest of the tokens under the learned normal-data conditionals. This observation motivates using masked reconstruction or conditional likelihood quantities derived from the masked diffusion model as anomaly scores in the next Section.

3 Methodology

3.1 MaskDiff-AD: Masked Diffusion for Anomaly Detection

Our proposed method consists of two steps. *(i) Probe masking:* given a test sample $\mathbf{x}$, we generate independent masked views $\widetilde{\mathbf{x}}^{(\ell,k)} \sim q_{\tau_\ell}(\cdot \mid \mathbf{x})$ at multiple masking levels $\tau_1, \ldots, \tau_L$. *(ii) Reconstruction scoring:* for each masked view, we use the trained masked diffusion model p_θ to compute the reconstruction surprisal of the masked values, and then average these scores across all views and masking levels. The framework is illustrated in Figure 1, and we detail these two steps below.

Probe masking. Let $0 < \tau_1 < \tau_2 < \cdots < \tau_L < 1$ be a pre-specified probe grid. At each probe level τ_ℓ, each coordinate of the test sample $\mathbf{x} \in \mathcal{X}$ is replaced by the mask m independently with the mask probability $1 - \alpha_{\tau_\ell}$ as shown in Eq. (1). This induces the following distribution of the masked view $\tilde{\mathbf{x}}$ under level τ_ℓ:

$$q_{\tau_\ell}(\widetilde{\mathbf{x}} \mid \mathbf{x}) := \prod_{j=1}^{d} \left[(1-\alpha_{\tau_\ell})\mathbf{1}\{\widetilde{x}_j = \mathsf{m}\} + \alpha_{\tau_\ell}\mathbf{1}\{\widetilde{x}_j = x_j\}\right], \tag{3}$$

which can be viewed as the special case of Eq. (1) under time $t = \tau_\ell$. For a masked view $\widetilde{\mathbf{x}}$, we write $M(\widetilde{\mathbf{x}}) := \{j : \widetilde{x}_j = \mathsf{m}\}$ and $V(\widetilde{\mathbf{x}}) := \{j : \widetilde{x}_j \neq \mathsf{m}\}$ for the masked and visible coordinate sets.

For a chosen integer K, we generate K independent probe-masked views for each probe level τ_ℓ as:

$$\widetilde{\mathbf{x}}^{(\ell,1)}, \ldots, \widetilde{\mathbf{x}}^{(\ell,K)} \overset{iid}{\sim} q_{\tau_\ell}(\cdot \mid \mathbf{x}), \text{ for } \ell = 1, 2, \ldots, L. \tag{4}$$

Larger values of τ_ℓ on average remove more information from the clean sample and therefore produce more heavily corrupted probe views. It is worthwhile noting that the probe levels $\tau_1, \ldots, \tau_L$ and the number of views K are hyperparameters that can be tuned for detection performance. In this work, we use a fixed, uniform probe grid in the main experiments for simplicity. We also perform sensitivity analyses on probe levels in Appendix B and C.

Reconstruction scoring. We propose to compute the anomaly score by evaluating *how difficult it is to recover the masked coordinates* of a test point under the predictive model trained on nominal data. Formally, denote the masked diffusion model trained from Eq. (2) as $p_{\hat{\theta}}^j(\cdot \mid \widetilde{\mathbf{x}})$, *i.e.*, the estimated conditional distribution of the j-th coordinate of the clean data. Then we define the single-probe reconstruction score as

$$s_{\mathrm{rec}}(\mathbf{x}; \widetilde{\mathbf{x}}) := -\frac{1}{|M(\widetilde{\mathbf{x}})| \vee 1} \sum_{j \in M(\widetilde{\mathbf{x}})} \log p_{\hat{\theta}}^j(x_j \mid \widetilde{\mathbf{x}}), \tag{5}$$

Algorithm 1 Test-Time MaskDiff-AD

Input: Test sample $\mathbf{x}$; pre-trained masked diffusion model $p_{\hat{\theta}}^{j}(\cdot \mid \mathbf{x}_t)$ for $j = 1, \ldots, d$; probe grids $\{\tau_\ell\}_{\ell=1}^{L}$; number of views per mask level K; calibrated detection threshold γ.
Output: Anomaly label $\hat{y}(\mathbf{x}) \in \{0, 1\}$, where 1 indicates anomalous and 0 indicates normal.
1: **for** each $\ell = 1, \ldots, L$ and $k = 1, \ldots, K$ **do**
2: Sample the corresponding probe-masked view $\widetilde{\mathbf{x}}^{(k,\ell)} \sim q_{\tau_\ell}(\cdot \mid \mathbf{x})$ according to Eq. (3).
3: Compute per-view reconstruction difficulty $s_{\mathrm{rec}}^{(k,\ell)}$ according to Eq. (5).
4: **end for**
5: Compute the aggregated anomaly score $S_{\mathrm{rec}}(\mathbf{x})$ according to Eq. (6).
6: **return** Anomaly label $\hat{y}(\mathbf{x}) = \mathbf{1}\{S_{\mathrm{rec}}(\mathbf{x}) > \gamma\}$.

i.e., the average surprisal of the true values at masked positions. Intuitively, a normal sample should have a lower reconstruction score, because its coordinates are consistent with the dependency structure learned from normal data and are therefore easier to predict from the visible context. In contrast, an anomalous sample is more likely to violate this dependency structure, making its masked coordinates harder to reconstruct and leading to a higher score, as visualized in Figure 1 and confirmed by the toy synthetic example in Figure 2.

To reduce sensitivity to a particular probe level or random mask, and to improve robustness across various anomalous scenarios, we average the reconstruction score in Eq. (5) over L probe levels and K independent masked views per level. Specifically, let $\{\widetilde{\mathbf{x}}^{(\ell,k)}\}_{\ell\in[L],\, k\in[K]}$ be sampled as in Eq. (4), the final aggregated anomaly score is

$$S_{\mathrm{rec}}(\mathbf{x}) := \frac{1}{LK} \sum_{\ell=1}^{L} \sum_{k=1}^{K} s_{\mathrm{rec}}\left(\mathbf{x};\, \widetilde{\mathbf{x}}^{(\ell,k)}\right). \tag{6}$$

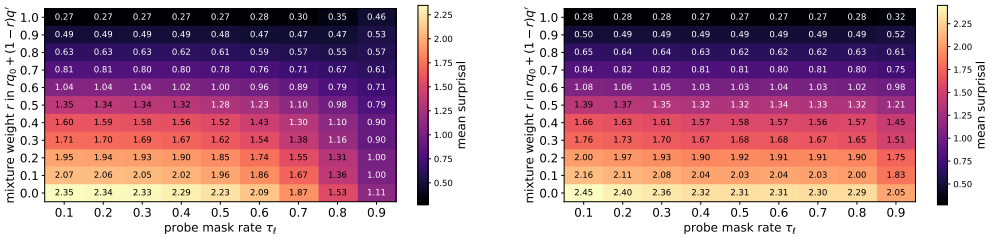
Anomaly detection is then performed by comparing $S_{\mathrm{rec}}(\mathbf{x})$ with a pre-specified threshold, where test data with larger scores are classified as anomalous. We refer to the resulting method as *Masked Diffusion for Anomaly Detection* (MaskDiff-AD). The full algorithm is summarized in Algorithm 1.

Remark 1 (Comparison with DTE)**.** *Our work is inspired by the diffusion time estimation (DTE) for anomaly detection (Livernoche et al., 2024), which computes the posterior of the diffusion time given the test sample and uses the posterior mean as the anomaly score. We would like to emphasize that DTE does not transfer naturally to MDM here. A natural attempt to extend DTE to MDM is to retain its corruption-time posterior, $q(t \mid \mathbf{x}) \propto q(t) q_t(\mathbf{x})$. In masked diffusion, however, q_t is defined on the extended space $\widetilde{\mathcal{X}}$ that includes the mask token* m*. If a clean sample $\mathbf{x} \in \mathcal{X}$ is viewed as an all-visible element of this space, then $q_t(\mathbf{x}) = \alpha_t^d q_0(\mathbf{x})$, so the resulting posterior $q(t \mid \mathbf{x}) \propto q(t)\alpha_t^d$ is identical for every clean test sample and therefore contains no discriminative signal. This motivated us to propose the reconstruction-based anomaly score instead under MDM.*

3.2 Some Variants of the Proposed Method

In this subsection, we discuss two practical variants of the proposed MaskDiff-AD framework. First, for small-scale categorical datasets, especially when many coordinates have binary or low-cardinality state spaces, we introduce a non-parametric version of the reconstruction score that estimates masked-coordinate conditional probabilities directly from normal training samples. This will avoid the training of parametric MDMs. Second, when a pre-trained MDM is unavailable and one does not need to re-train a generic MDM, we provide a slightly different alternative loss to Eq. (2), for training the predictive models $p_{\theta}^{j}(\cdot \mid \widetilde{\mathbf{x}})$. This objective is more aligned with our anomaly score and can be useful especially for single-probe scoring.

Non-parametric reconstruction score. For a probe-masked view $\widetilde{\mathbf{x}}$, instead of training a predictive parametric model $\{p_{\hat{\theta}}^{j}(\cdot \mid \mathbf{x}_t), j = 1, \ldots, d\}$ to compute the anomaly score, we can also adopt the kernel-smoothed empirical conditional distribution as follows. Recall that $M(\widetilde{\mathbf{x}})$ and $V(\widetilde{\mathbf{x}})$ denote its masked and visible coordinate sets, respectively. Specifically, for each normal training sample



(a) Non-parametric score $\mathbb{E}[-\log \hat{p}^j_{\mathrm{NP}}(x_j \mid \widetilde{\mathbf{x}})]$ (b) Parametric score $\mathbb{E}[-\log p^j_{\hat{\theta}}(x_j \mid \widetilde{\mathbf{x}})]$

Figure 2: Synthetic heatmaps of the expectation of non-parametric and parametric reconstruction anomaly score under the mixture $\mu_r = rq_0 + (1-r)q'$. Here, q_0 denotes the normal distribution over length-40 binary vectors, generated by drawing $S \sim \mathrm{Bernoulli}(0.5)$ and making each coordinate a noisy copy of S with an 8% bit-flip probability. The anomaly distribution q' follows the same construction except that the target coordinate j is flipped relative to S, preserving its marginal while changing its conditional relationship with the visible context. For each $\mathbf{x} \sim \mu_r$, we mask coordinates at rate $1 - \alpha_{\tau_\ell}$ and then force the target coordinate j to be masked.

$\mathbf{x}^{(n)} \in \mathcal{D}_{\mathrm{tr}}$, define the visible-coordinate Hamming distance between $\mathbf{x}^{(n)}$ and $\widetilde{\mathbf{x}}$ as

$$d_{\mathrm{vis}}(\widetilde{\mathbf{x}}, \mathbf{x}^{(n)}) := \sum_{r \in V(\widetilde{\mathbf{x}})} \mathbf{1}\{\widetilde{x}_r \neq x_r^{(n)}\}.$$

Then we estimate the reconstruction probability by

$$\widehat{p}^j_{\mathrm{NP}}(x_j = a \mid \widetilde{\mathbf{x}}) := \frac{\sum_{n=1}^N \mathbf{1}\{x_j^{(n)} = a\}\, K_\lambda(\widetilde{\mathbf{x}}, \mathbf{x}^{(n)})}{\sum_{n=1}^N K_\lambda(\widetilde{\mathbf{x}}, \mathbf{x}^{(n)})}, \quad \forall j \in M(\widetilde{\mathbf{x}}), a \in \mathcal{X}_j. \tag{7}$$

Here $K_\lambda(\widetilde{\mathbf{x}}, \mathbf{x}^{(n)}) = \exp\big(-\lambda d_{\mathrm{vis}}(\widetilde{\mathbf{x}}, \mathbf{x}^{(n)})\big)$ is the kernel weight and $\lambda > 0$ is a pre-specified bandwidth parameter. The non-parametric anomaly score can be computed similarly as in Algorithm 1 by replacing the parametric predictive model $p^j_{\hat{\theta}}(\cdot \mid \mathbf{x}_t)$ with the non-parametric model in Eq. (7). The algorithm is given in Algorithm 2 in the Appendix for completeness.

To illustrate the non-parametric MaskDiff-AD score and compare it with the parametric score in Eq. (6), we evaluate both on a synthetic setting with varying anomaly levels in Figure 2, where $r = 0$ (bottom in each heatmap) corresponds to fully anomalous samples and $r = 1$ (top in each heatmap) to fully nominal samples. As samples move away from the normal distribution, the reconstruction posterior assigned to the true masked value decreases and the corresponding surprisal increases. This implies that both parametric and non-parametric scores can identify anomalous samples; the non-parametric score is comparable at smaller mask rates, while the parametric score becomes more effective at larger mask rates. For small datasets with simple discrete structure, however, the non-parametric version may be preferable because it requires no model training.

An alternative training objective. When a pre-trained MDM is available, MaskDiff-AD can be applied directly using the trained model without additional training. When such a model is unavailable and the goal is anomaly detection, it is not always necessary to train a generic MDM over all diffusion times. Instead, it suffices to learn the conditional distributions needed for the reconstruction score at the probe levels used at test time. To this end, we may also choose to train a time-conditioned predictive model $p^j_{\theta_{\mathrm{rec}}}(\cdot \mid \widetilde{\mathbf{x}}, \ell)$, which estimates the distribution of the clean coordinate x_j given a masked view $\widetilde{\mathbf{x}}$ and the probe-level index ℓ. Specifically, for each normal training sample $\mathbf{x}^{(n)} \in \mathcal{D}_{\mathrm{tr}}$ and each probe level τ_ℓ, we draw a masked view $\widetilde{\mathbf{x}}^{(\ell,n)} \sim q_{\tau_\ell}(\cdot \mid \mathbf{x}^{(n)})$. We then fit the predictive model by minimizing the average masked reconstruction loss

$$\mathcal{L}_{\mathrm{rec}}(\theta_{\mathrm{rec}}) = -\frac{1}{NL} \sum_{n=1}^N \sum_{\ell=1}^L \frac{1}{|M(\widetilde{\mathbf{x}}^{(\ell,n)})| \vee 1} \sum_{j \in M(\widetilde{\mathbf{x}}^{(\ell,n)})} \log p^j_{\theta_{\mathrm{rec}}}\big(x_j^{(n)} \mid \widetilde{\mathbf{x}}^{(\ell,n)}, \ell\big). \tag{8}$$

This objective directly matches the test-time reconstruction score: the model is trained to predict only the coordinates that are masked in the probe views, and the loss is normalized per view to make different masking levels comparable. Averaging over $\ell = 1, \ldots, L$ trains the model for the

full multi-probe score used in Eq. (6); for single-probe scoring, one can set $L = 1$ and train only at the chosen probe level. In this sense, the objective is tailored to anomaly scoring rather than to general-purpose masked diffusion generation.

The training loss in Eq. (8) closely resembles that of MDM in Eq. (2), with two key distinctions. First, our $\mathcal{L}_{\text{rec}}$ is normalized by its realized number of masked coordinates, whereas this normalization is replaced with a time-dependent constant $w(t) = \alpha'_t/(1-\alpha_t)$ in the standard MDM loss. Second, the loss in Eq. (8) is only computed with the fixed L mask levels that will be used in detection. This seemingly minor difference reflects the distinct objectives of the two settings. For sampling, Sahoo et al. (2024); Shi et al. (2024) showed that the MDM loss admits a clear interpretation as the KL divergence between probability paths, and minimizing it ensures effective generation along the diffusion trajectory. In contrast, for anomaly detection, our normalized $\mathcal{L}_{\text{rec}}$ is directly aligned with the test-time scoring used in MaskDiff-AD. Furthermore, the loss $\mathcal{L}_{\text{rec}}$ is expected to be trained exclusively on nominal data, while a pre-trained MDM might be trained on a more diverse dataset as in general-purpose language modeling.

3.3 Performance Guarantees

In this subsection, we provide some theoretical insights for the Type-I and Type-II errors of the proposed algorithm under a well-specified reconstruction model. Let $q^j(\cdot \mid \widetilde{\mathbf{x}}, \tau_\ell)$ denote the oracle conditional distribution at probe level τ_ℓ, and let the corresponding oracle reconstruction score be

$$S^*(\mathbf{x}) := \frac{1}{L}\sum_{\ell=1}^{L} \mathbb{E}_{\widetilde{\mathbf{x}}\sim q_{\tau_\ell|0}(\cdot|\mathbf{x})}\Big(-\frac{1}{|M(\widetilde{\mathbf{x}})| \vee 1} \sum_{j\in M(\widetilde{\mathbf{x}})} \log q^j\big(x_j \mid \widetilde{\mathbf{x}}, \tau_\ell\big)\Big). \tag{9}$$

We also define

$$S^*_{\text{rec}}(\mathbf{x}) := \frac{1}{L}\sum_{\ell=1}^{L} \mathbb{E}_{\widetilde{\mathbf{x}}\sim q_{\tau_\ell|0}(\cdot|\mathbf{x})}\Big(-\frac{1}{|M(\widetilde{\mathbf{x}})| \vee 1} \sum_{j\in M(\widetilde{\mathbf{x}})} \log p^j_{\hat{\theta}}\big(x_j \mid \widetilde{\mathbf{x}}, \tau_\ell\big)\Big), \tag{10}$$

which is the same quantity as $S^*(\mathbf{x})$ except when the reconstruction model is learned. Finally, we let $S_{\text{rec}}(\mathbf{x})$ be the anomaly score in Eq. (6), which is a sampled version (or realization) of $S^*_{\text{rec}}(\mathbf{x})$. The following theorem summarizes the guarantee on Type-I/Type-II errors of the anomaly detector. The proof can be found in Appendix D.

Theorem 1. *We assume there exists constant $C_\ell > 0$ for probe level τ_ℓ such that for all j and all masked views $\widetilde{\mathbf{x}}$, the trained and true models satisfy $-\log p^j_{\widehat{\theta}_{\text{rec}}}(x_j \mid \widetilde{\mathbf{x}}, \tau_\ell) \in [0, C_\ell]$, and $-\log q^j(x_j \mid \widetilde{\mathbf{x}}, \tau_\ell) \in [0, C_\ell]$. Let $\overline{C} = \frac{1}{L}\sum_{\ell=1}^{L} C_\ell$ and $C^2_{\text{rms}} = \frac{1}{L}\sum_{\ell=1}^{L} C^2_\ell$. We also assume that the training loss is small and define the following one-sided misspecification errors*

$$\epsilon_0 := \mathbb{E}_{\mathbf{x}\sim q_0}\left[S^*_{\text{rec}}(\mathbf{x}) - S^*(\mathbf{x})\right], \qquad \epsilon_1 := \left[\mathbb{E}_{\mathbf{x}\sim q'}\left[S^*(\mathbf{x}) - S^*_{\text{rec}}(\mathbf{x})\right]\right]_+.$$

Then, for a fixed threshold γ, we have:

- *Type-I error: for normal data $\mathbf{x} \sim q_0$, let $\mu^*_0 := \mathbb{E}_{\mathbf{x}\sim q_0}[S^*(\mathbf{x})]$. Assume $\gamma > \mu^*_0 + \epsilon_0$, then*

$$\mathbb{P}_{\mathbf{x}\sim q_0}(S_{\text{rec}}(\mathbf{x}) > \gamma) \le \exp\left(-\frac{LK(\gamma - \mu^*_0 - \epsilon_0)^2}{2C^2_{\text{rms}}}\right) + \exp\left(-\frac{(\gamma - \mu^*_0 - \epsilon_0)^2}{2\overline{C}^2}\right).$$

- *Type-II error: for anomalous data $\mathbf{x} \sim q'$, let $\mu^*_1 := \mathbb{E}_{q'}[S^*(\mathbf{x})]$. Assume that the threshold satisfies $\mu^*_0 + \epsilon_0 < \gamma < \mu^*_1 - \epsilon_1$. Then,*

$$\mathbb{P}_{\mathbf{x}\sim q'}(S_{\text{rec}}(\mathbf{x}) \le \gamma) \le \exp\left(-\frac{LK(\mu^*_1 - \epsilon_1 - \gamma)^2}{2C^2_{\text{rms}}}\right) + \exp\left(-\frac{(\mu^*_1 - \epsilon_1 - \gamma)^2}{2\overline{C}^2}\right).$$

Remark 2. *We further discuss the gap between μ^*_0 and μ^*_1. For a probe level τ_ℓ, let M and V be the data-independent masked and visible sets induced by Eq. (3). Write $q^j(\cdot \mid x_V)$ and $q'^j(\cdot \mid x_V)$ for the conditional distributions of X_j given $X_V = x_V$ under q_0 and q', respectively. The oracle mean scores are $\mu^*_{0,\ell} := \mathbb{E}_M\mathbb{E}_{\mathbf{x}\sim q_0}[-\frac{1}{|M|\vee 1}\sum_{j\in M}\log q^j(x_j \mid x_V)]$ and $\mu^*_{1,\ell} := \mathbb{E}_M\mathbb{E}_{\mathbf{x}\sim q'}[-\frac{1}{|M|\vee 1}\sum_{j\in M}\log q^j(x_j \mid x_V)]$. Define the average conditional KL-divergence*

$$D_\ell := E_M\Big[\frac{1}{|M| \vee 1}\sum_{j\in M} E_{X_V\sim q'_V} KL\big(q'^j(\cdot|X_V)\|q^j(\cdot|X_V)\big)\Big].$$

Table 1: Average ranks and mean ROC-AUC/PR-AUC over 14 tabular datasets and five random seeds. Overall rank averages the ROC-AUC and PR-AUC ranks.

Method	Overall Rank	ROC-AUC Rank	PR-AUC Rank	Mean ROC-AUC	Mean PR-AUC
Parametric MaskDiff-AD	**3.929**	**3.714**	**4.143**	**80.66**	32.39
Non-parametric MaskDiff-AD	4.929	4.857	5.000	79.12	31.64
DTE-Gaussian	5.286	4.929	5.643	77.29	**35.34**
Hamming-kNN	6.250	6.000	6.500	75.66	27.02
DTE-Categorical	6.357	5.214	7.500	78.86	25.79
DRL	7.429	7.857	7.000	71.28	29.10
ECOD	7.857	7.429	8.286	75.08	24.87
COPOD	7.929	7.857	8.000	75.30	24.92
GOAD	8.464	8.500	8.429	73.17	24.73
DTE-InvGamma	8.571	9.429	7.714	74.38	29.17
ICL	8.786	9.286	8.286	60.95	23.67
MCM	8.821	9.214	8.429	72.69	24.41
DeepSVDD	9.679	10.786	8.571	68.22	29.49
IForest	10.714	9.929	11.500	68.40	15.45

Let H'_ℓ and H_ℓ denote the corresponding average conditional entropies under q' and q_0. Then the oracle separation at level τ_ℓ is

$$\mu^*_{1,\ell} - \mu^*_{0,\ell} = D_\ell + H'_\ell - H_\ell.$$

This is positive as long as $D_\ell > H_\ell - H'_\ell$. The case with multi-probe score is simply the corresponding average over ℓ. This gives us a quantitative description of the separation. Intuitively, when the anomaly changes conditional dependencies while leaving the conditional entropies approximately unchanged, the score gap is approximately the average conditional KL divergence.

4 Numerical Experiments

Numerical setup. We evaluate MaskDiff-AD on *eighteen* real-world datasets, including *thirteen* fully discrete tabular data, *one* mixed-type tabular data (with both categorical variables and continuous variables), and *four* discrete text sequences. The summary of datasets is shown in Table 2 in Appendix B.1. All experiments follow the normal-only training of the anomaly detection protocol: during training, each method has access only to samples from the normal class; anomalous samples are used only for test-time evaluation. On tabular data, features are represented as discrete symbols, with continuous features discretized when necessary. On text data, each document is converted into a fixed-length token sequence and masked reconstruction is performed at the token level. This allows the same forward-only scoring principle to be applied across different discrete data modalities.

For MaskDiff-AD, we use fixed hyperparameters across datasets rather than validation-based or per-dataset tuning. We use the standard linear absorbing schedule $\alpha_t = 1 - t$, so the probe level τ_ℓ has mask probability $1 - \alpha_{\tau_\ell} = \tau_\ell$. The parametric tabular and text models use the uniform probe grid $\tau_\ell \in \{0.1, 0.2, \ldots, 0.9\}$, while the non-parametric tabular model uses $\tau_\ell \in \{0.15, 0.30, 0.45, 0.60\}$. For tabular data, the parametric reconstruction score averages over $K = 16$ masked probe views at each probe level. The non-parametric model uses the visible-coordinate Hamming kernel with bandwidth $\lambda = 1$, takes all normal training samples as the reference set, and averages over 8 masked probe views at each probe level. For text data, we use the GPT-2 (Radford et al., 2019) tokenizer only, without using pretrained GPT-2 weights, and we average the reconstruction score over 24 independently generated masked probe views at each probe level. Model architectures, optimization details, and additional implementation hyperparameters are provided in Appendix B.3.

Baselines and metrics. We compare MaskDiff-AD with classical anomaly detection methods, DTE-based variants (Livernoche et al., 2024), and recent tabular anomaly detectors, including MCM (Yin et al., 2024), DRL (Ye et al., 2025), NPT-AD (Thimonier et al., 2024), GNSM (Mahmood et al., 2024), and AnoLLM (Tsai et al., 2025). For text data, we compare against DATE (Manolache et al., 2021), DTE variants, and anomaly detectors applied to fixed BERT (Devlin et al., 2019) or OpenAI (OpenAI, 2026) embeddings. We report ROC-AUC and PR-AUC. Complete baseline descriptions, implementation details, and evaluation protocols are provided in Appendix B.4.

Results. Table 1, summarized from the results in Tables 10 and 11 in the Appendix, shows that our reconstruction-based variants are overall the strongest on tabular data. In particular, the parametric reconstruction anomaly score achieves the best average overall rank (3.929), and the highest mean

ROC-AUC (80.66), while the non-parametric variant ranks second overall. DTE-Gaussian is the strongest baseline by overall average rank and achieves the highest mean PR-AUC. Overall, the results suggest that reconstruction-based masked conditional scoring provides strong and consistent performance across heterogeneous categorical and mixed-type datasets.

We further examine whether the comparison is sensitive to the representation used by continuous-space baselines. Since the DTE variants operate on continuous embeddings of the discrete features, we additionally evaluate them using a 32-dimensional PCA representation fitted only on nominal training data; such compact representation does not improve their mean ROC-AUC or PR-AUC over the one-hot representation (Appendix C.2). We also evaluate continuous-input baselines on Vehicle Claims using the original numerical features rather than their discretized versions, and Parametric MaskDiff-AD remains the strongest method under both ROC-AUC and PR-AUC (Appendix C.3). These results indicate that the observed advantage of MaskDiff-AD is not attributable solely to the particular discretization or continuous embedding used in the main comparison.

For text data, Table 12 shows that the proposed parametric reconstruction score performs particularly well on the two spam-detection datasets, achieving the best ROC-AUC and PR-AUC on both Email Spam and SMS Spam and outperforming DATE as well as the embedding-based baselines. On AGNews and YelpReview, however, our method is less competitive than the strongest OpenAI-embedding baselines. Since these baselines rely on representations pretrained on large-scale external corpora, whereas our model is trained from scratch using only nominal samples from each dataset, the performance gap may reflect differences in representation quality, task semantics, and the capacity of the current token-level architecture. Overall, the results suggest that masked reconstruction provides a promising anomaly signal for discrete text sequences, while also motivating the use of stronger pretrained or text-specific diffusion backbones for semantically richer text anomaly detection tasks.

Sensitivity analysis on probe level τ. We further study the effect of the probe level using the Vehicle Claims dataset. Figure 3 reports the ROC-AUC of the proposed parametric score when computed from a single probe mask rate τ. The ROC-AUC first increases at small τ and then decreases when too much context is removed. This suggests a trade-off: at small τ the algorithm masks too few coordinates to expose anomalous dependencies, while at large τ the missing context makes reconstruction difficult for both normal and anomalous samples. Additional results in Appendix B.5 show that the best single probe level can differ across datasets. Therefore, we use a fixed multi-probe score in the main experiments, aggregating over a uniform grid of probe levels instead of tuning τ separately for each dataset.

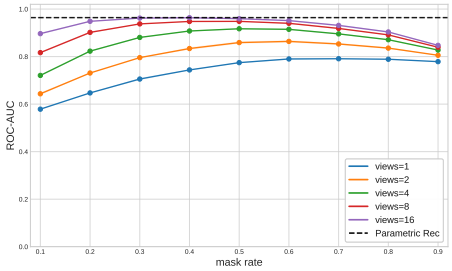


Figure 3: Sensitivity of the ROC-AUC to the probe mask rate on the Vehicle Claims dataset. Each solid curve uses a single mask rate and averages over $K \in \{1, 2, 4, 8, 16\}$ masked views. The dashed line denotes our default multi-probe parametric reconstruction score.

5 Conclusion and Discussions

We proposed MaskDiff-AD, a masked diffusion-based anomaly detection framework for discrete data. The method uses a trained masked diffusion model for anomaly scoring: a test sample is randomly masked at multiple probe levels, and its anomaly score is the average reconstruction surprisal of the masked entries. This design directly leverages the discrete structure of the data and is computationally efficient to compute. Our theoretical analysis provided Type-I and Type-II error insights under well-specified reconstruction models, and our experiments showed competitive performance across eighteen tabular and text datasets. One limitation is that the method depends on the quality of the learned masked conditionals and on the choice of probe levels, although multi-probe averaging helps to reduce this sensitivity. In addition, its performance on long text sequences is currently less competitive, which may be partly due to the simple text modeling and token-level architecture used in our experiments. Future work includes adaptive selection of probe levels, text-specific architectures and training strategies for more semantically complex text domains, extensions to structured discrete objects such as graphs and molecules, and tighter theory connecting masked reconstruction with distributional divergences between nominal and anomalous data.

AI USE STATEMENT

In this work, we used generative AI tools solely for writing polish and coding assistance. The methods, theoretical analysis, and proofs were developed entirely by the authors without generative AI assistance. All AI-assisted text and code were reviewed and verified by the authors. We take responsibility for the final content of this work, including text or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To ensure reproducibility, we describe the proposed method and provide algorithmic pseudocode in Section 3 and Appendix A, with source code included in the supplementary materials. The theoretical assumptions and guarantees are presented in Section 3.3, with proofs and further discussion in Appendix D. Appendix B documents the datasets, preprocessing procedures, train/test splits, evaluation metrics, model architectures, training hyperparameters, and baseline implementations. Additional comparisons and sensitivity analyses are provided in Appendix C, and detailed experimental results across multiple random seeds are reported in Appendix E.

REFERENCES

Charu C. Aggarwal. *Outlier Analysis*. Springer, 2 edition, 2017. doi: 10.1007/978-3-319-47578-3.

Mohiuddin Ahmed, Abdun Naser Mahmood, and Md Rafiqul Islam. A survey of anomaly detection techniques in financial domain. *Future Generation Computer Systems*, 55:278–288, 2016.

Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian conference on computer vision*, pp. 622–637. Springer, 2018.

Leman Akoglu, Hanghang Tong, Jilles Vreeken, and Christos Faloutsos. Fast and reliable anomaly detection in categorical data. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pp. 415–424, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450311564. doi: 10.1145/2396761.2396816. URL https://doi.org/10.1145/2396761.2396816.

Jinwon An and Sungzoon Cho. Variational autoencoder based anomaly detection using reconstruction probability. *Special lecture on IE*, 2(1):1–18, 2015.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021.

Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. In *International Conference on Learning Representations (ICLR)*, 2020.

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

Michael Cardei, Jacob K Christopher, Thomas Hartvigsen, Bhavya Kailkhura, and Ferdinando Fioretto. Constrained discrete diffusion. *arXiv preprint arXiv:2503.09790*, 2025.

Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):1–58, 2009.

Ajay Chawda, Stefanie Grimm, and Marius Kloft. Unsupervised anomaly detection for auditing data and impact of categorical encodings. *arXiv preprint arXiv:2210.14056*, 2022.

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

Katherine Fraser, Samuel Homiller, Rashmish K Mishra, Bryan Ostdiek, and Matthew D Schwartz. Challenges for unsupervised anomaly detection in particle physics. *Journal of High Energy Physics*, 2022(3):66, 2022.

Adam Goodge, Bryan Hooi, See-Kiong Ng, and Wee Siong Ng. Lunar: Unifying local outlier detection methods via graph neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(6):6737–6745, Jun. 2022. doi: 10.1609/aaai.v36i6.20629. URL https://ojs.aaai.org/index.php/AAAI/article/view/20629.

Songqiao Han, Xiyang Hu, Hailiang Huang, Minqi Jiang, and Yue Zhao. Adbench: Anomaly detection benchmark. *Advances in Neural Information Processing Systems*, 35:32142–32159, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. doi: 10.48550/arXiv.1312.6114.

Yuangang Li, Jiaqi Li, Zhuo Xiao, Tiankai Yang, Yi Nian, Xiyang Hu, and Yue Zhao. Nlp-adbench: Nlp anomaly detection benchmark. *Findings of the Association for Computational Linguistics: EMNLP 2025*, 2025.

Zheng Li, Yue Zhao, Nicola Botta, Cezar Ionescu, and Xiyang Hu. Copod: Copula-based outlier detection. In *2020 IEEE International Conference on Data Mining (ICDM)*, pp. 1118–1123. IEEE, November 2020. doi: 10.1109/icdm50108.2020.00135. URL http://dx.doi.org/10.1109/ICDM50108.2020.00135.

Zheng Li, Yue Zhao, Xiyang Hu, Nicola Botta, Cezar Ionescu, and George H. Chen. ECOD: Unsupervised outlier detection using empirical cumulative distribution functions. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12181–12193, 2023. doi: 10.1109/TKDE.2022.3159580.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pp. 413–422, USA, 2008. IEEE Computer Society. ISBN 9780769535029. doi: 10.1109/ICDM.2008.17. URL https://doi.org/10.1109/ICDM.2008.17.

Jing Liu, Zhenchao Ma, Zepu Wang, Chenxuanyin Zou, Jiayang Ren, Zehua Wang, Liang Song, Bo Hu, Yang Liu, and Victor Leung. A survey on diffusion models for anomaly detection. *arXiv preprint arXiv:2501.11430*, 2025.

Yezheng Liu, Zhe Li, Chong Zhou, Yuanchun Jiang, Jianshan Sun, Meng Wang, and Xiangnan He. Generative adversarial active learning for unsupervised outlier detection. *IEEE Transactions on Knowledge and Data Engineering*, 32(8):1517–1528, 2019.

Victor Livernoche, Vineet Jain, Yashar Hezaveh, and Siamak Ravanbakhsh. On diffusion modeling for anomaly detection. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=lR3rk7ysXz.

Ahsan Mahmood, Junier Oliva, and Martin Andreas Styner. Anomaly detection via gumbel noise score matching. *Frontiers in Artificial Intelligence*, Volume 7 - 2024, 2024. ISSN 2624-8212. doi: 10.3389/frai.2024.1441205. URL https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2024.1441205.

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

Arian Mousakhan, Thomas Brox, and Jawad Tayyub. Anomaly detection with conditioned denoising diffusion models. In *DAGM German Conference on Pattern Recognition*, pp. 181–195. Springer, 2024.

OpenAI. New embedding models and api updates. https://openai.com/index/new-embedding-models-and-api-updates/, 2026.

Girik Pachauri and Sandeep Sharma. Anomaly detection in medical wireless sensor networks using machine learning algorithms. *Procedia Computer Science*, 70:325–333, 2015.

Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM Computing Surveys (CSUR)*, 54(2):1–38, 2021.

Mihir Prabhudesai, Mengning Wu, Amir Zadeh, Katerina Fragkiadaki, and Deepak Pathak. Diffusion beats autoregressive in data-constrained settings. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=W5Ht05jF4c.

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

Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. Efficient algorithms for mining outliers from large data sets. *SIGMOD Record*, 29(2):427–438, June 2000. ISSN 0163-5808. doi: 10.1145/335191.335437.

Marco Rudolph, Bastian Wandt, and Bodo Rosenhahn. Same same but differnet: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 1907–1916, 2021.

Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4393–4402. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/ruff18a.html.

Lukas Ruff, Jacob R Kauffmann, Robert A Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G Dietterich, and Klaus-Robert Müller. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 109(5):756–795, 2021.

Subham S Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024.

Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*, pp. 4–11, 2014.

Osman Salem, Alexey Guerassimov, Ahmed Mehaoua, Anthony Marcus, and Borko Furht. Sensor fault and patient anomaly detection and classification in medical wireless sensor networks. In *2013 IEEE International Conference on Communications (ICC)*, pp. 4373–4378. IEEE, 2013.

Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *Information Processing in Medical Imaging*, pp. 146–157. Springer, 2017.

Tom Shenkar and Lior Wolf. Anomaly detection for tabular data with internal contrastive learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=_hszZbt46bT.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=xcqSOfHt4g.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. pmlr, 2015.

Gian Antonio Susto, Matteo Terzi, and Alessandro Beghi. Anomaly detection approaches for semiconductor manufacturing. *Procedia Manufacturing*, 11:2018–2024, 2017.

Ayman Taha and Ali S Hadi. Anomaly detection methods for categorical data: A review. *ACM Computing Surveys (CSUR)*, 52(2):1–35, 2019.

Hugo Thimonier, Fabrice Popineau, Arpad Rimmel, and Bich-Liên Doan. Beyond individual input for deep anomaly detection on tabular data. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 48097–48123. PMLR, 21–27 Jul 2024.

Che-Ping Tsai, Ganyu Teng, Phil Wallis, and Wei Ding. Anollm: Large language models for tabular anomaly detection. In *The thirteenth International Conference on Learning Representations*, 2025. Accepted, to appear.

Masatoshi Uehara, Yulai Zhao, Tommaso Biancalani, and Sergey Levine. Understanding reinforcement learning-based fine-tuning of diffusion models: A tutorial and review. *arXiv preprint arXiv:2407.13734*, 2024.

Julian Wyatt, Adam Leach, Sebastian M Schmon, and Chris G Willcocks. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 650–656, 2022.

Takehisa Yairi, Yoshinobu Kawahara, Ryohei Fujimaki, Yuichi Sato, and Kazuo Machida. Telemetry-mining: a machine learning approach to anomaly detection and fault diagnosis for space systems. In *2nd IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT'06)*, pp. 8–pp. IEEE, 2006.

Hangting Ye, He Zhao, Wei Fan, Mingyuan Zhou, Dan dan Guo, and Yi Chang. DRL: Decomposed representation learning for tabular anomaly detection. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=CJnceDksRd.

Jiaxin Yin, Yuanyuan Qiao, Zitang Zhou, Xiangchao Wang, and Jie Yang. MCM: Masked cell modeling for anomaly detection in tabular data. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=lNZJyEDxy4.

Hui Zhang, Zheng Wang, Dan Zeng, Zuxuan Wu, and Yu-Gang Jiang. DiffusionAD: Norm-guided one-step denoising diffusion for anomaly detection. *IEEE transactions on pattern analysis and machine intelligence*, 2025.

Kaiwen Zheng, Yongxin Chen, Hanzi Mao, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. In *International Conference on Learning Representations (ICLR)*, 2025.

Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding Gaussian mixture model for unsupervised anomaly detection. In *International Conference on Learning Representations*, 2018.

A Algorithmic Details

In this section, we provide the detailed algorithms for the two variants mentioned in Section 3.2. Algorithm 2 summarizes the non-parametric version of the MaskDiff-AD using the non-parametric score computed from Eq. (7). Algorithm 3 summarizes the training procedure when using the alternative training objective in Eq. (8).

Algorithm 2 Non-Parametric MaskDiff-AD

Input: Test sample $\mathbf{x}$; normal-only training data $\mathcal{D}_{\text{tr}}$; probe grids $\{\tau_\ell\}_{\ell=1}^{L}$; number of views per mask level K; detection threshold γ.
Output: Anomaly label $\hat{y}(\mathbf{x}) \in \{0, 1\}$, where 1 indicates anomalous and 0 indicates normal.

1: **for** each $\ell = 1, \ldots, L$ and $k = 1, \ldots, K$ **do**
2: Sample the corresponding probe-masked view $\widetilde{\mathbf{x}}^{(k,\ell)} \sim q_{\tau_\ell|0}(\cdot \mid \mathbf{x})$ according to Eq. (3).
3: Compute per-view reconstruction difficulty

$$s_{\text{rec}}^{\text{NP}}\big(\mathbf{x}; \widetilde{\mathbf{x}}^{(\ell,k)}\big) = -\frac{1}{|M(\widetilde{\mathbf{x}}^{(\ell,k)})| \vee 1} \sum_{j \in M(\widetilde{\mathbf{x}}^{(\ell,k)})} \log \widehat{p}_{\text{NP}}^{j}(x_j \mid \widetilde{\mathbf{x}}^{(\ell,k)})$$

where $\widehat{p}_{\text{NP}}^{j}(\cdot \mid \widetilde{\mathbf{x}})$ is as in Eq. (7).
4: **end for**
5: Compute the aggregated anomaly score

$$S_{\text{rec}}^{\text{NP}}(\mathbf{x}) := \frac{1}{LK} \sum_{\ell=1}^{L} \sum_{k=1}^{K} s_{\text{rec}}^{\text{NP}}\big(\mathbf{x}; \widetilde{\mathbf{x}}^{(\ell,k)}\big).$$

6: **return** Anomaly label $\hat{y}(\mathbf{x}) = \mathbf{1}\{S_{\text{rec}}^{\text{NP}}(\mathbf{x}) > \gamma\}$.

Algorithm 3 Training of Alternative Parametric Reconstruction Model

Input: Normal training set $D_{\text{tr}} = \{\mathbf{x}^{(n)}\}_{n=1}^{N}$, probe grids rates $\{\tau_\ell\}_{\ell=1}^{L}$, total epochs E, learning rate η.
Output: Trained reconstruction model $p_{\widehat{\theta}_{\text{rec}}}^{j}(\cdot \mid \widetilde{\mathbf{x}}), j = 1, \ldots, d$.

1: Initialize reconstruction model $p_{\widehat{\theta}_{\text{rec}}}^{j}$
2: **for** epoch $= 1, \ldots, E$ **do**
3: **for** each mask level $\ell = 1, \ldots, L$ **do**
4: Sample masked views $\widetilde{\mathbf{x}}^{(\ell,n)} \sim q_{\tau_\ell|0}(\cdot \mid \mathbf{x}^{(n)}), n = 1, \ldots, N$
5: **end for**
6: Compute the reconstruction loss $\mathcal{L}_{\text{rec}}(\theta_{\text{rec}})$ according to (8)
7: Update parameters $\theta_{\text{rec}} \leftarrow \theta_{\text{rec}} - \eta \nabla_{\theta_{\text{rec}}} \mathcal{L}_{\text{rec}}(\theta_{\text{rec}})$
8: **end for**

B Experiment Details

B.1 Datasets and Pre-processing

Table 2 summarizes the eighteen datasets we used for evaluating the proposed method. The datasets include thirteen categorical tabular datasets from ADRepository (Pang et al., 2021), one mixed-type vehicle claims dataset from UADAD (Chawda et al., 2022), and four text anomaly detection datasets from NLP-ADBench (Li et al., 2025). All experiments follow the normal-only anomaly detection protocol: anomaly labels are used only for constructing stratified splits and for test-time evaluation, but are never used for training the proposed method.

UADAD. We use the labeled vehicle claims dataset from the UADAD benchmark (Chawda et al., 2022). Following the main feature setting of that benchmark, we retain the following attributes:

`Maker`, `Genmodel`, `Color`, `Reg_year`, `Bodytype`, `Runned_Miles`, `Engin_size`, `Gearbox`, `Fuel_type`, `Price`, `Seat_num`, `Door_num`, `issue`, `issue_id`, `repair_complexity`, `repair_hours`, `repair_cost`.

The column `Label` is used only for splitting and evaluation, where `Label` $= 0$ denotes a normal sample and `Label` $= 1$ denotes an anomalous sample. We randomly split the full labeled dataset into 70% training and 30% testing using a stratified split with respect to `Label`. To match the unsupervised anomaly detection setting, only normal samples with `Label` $= 0$ in the training split

Table 2: Datasets overview.

Dataset Name	# Samples	# Features	% Anomaly	Data Type
AD (Pang et al., 2021)	3,279	1,555	14.00%	Categorical
AID362 (Pang et al., 2021)	4,279	114	1.40%	Categorical
APAS (Pang et al., 2021)	12,695	64	1.39%	Categorical
bank (Pang et al., 2021)	41,188	10	11.27%	Categorical
census (Pang et al., 2021)	299,285	33	6.20%	Categorical
Chess (Pang et al., 2021)	28,056	6	0.10%	Categorical
CMC (Pang et al., 2021)	1,473	8	1.97%	Categorical
CoverType (Pang et al., 2021)	581,012	44	0.47%	Categorical
probe (Pang et al., 2021)	64,759	6	6.43%	Categorical
R10 (Pang et al., 2021)	12,897	100	1.84%	Categorical
Solar (Pang et al., 2021)	1,066	11	4.03%	Categorical
U2R (Pang et al., 2021)	60,821	6	0.38%	Categorical
Vehicle claims (Chawda et al., 2022)	268,255	17	21.15%	Mixed-type
w7a (Pang et al., 2021)	49,749	300	2.97%	Categorical
AGNews (Li et al., 2025)	98,207	128	3.85%	NLP
Email spam (Li et al., 2025)	3,578	256	4.08%	NLP
SMS spam (Li et al., 2025)	4,672	128	3.30%	NLP
YelpReview (Li et al., 2025)	316,924	256	5.66%	NLP

are used to train the model or build the reference set. The full test split, containing both normal and anomalous samples, is used for evaluation.

We treat the vehicle claims dataset as a mixed-type tabular dataset. Categorical attributes are kept as discrete symbols and encoded into integer IDs. Numerical attributes are discretized by quantile binning, where the bin boundaries are fitted using only the normal training samples and then applied to the test set. This yields a fully discrete representation for MaskDiff-AD and the corresponding discrete baselines.

ADRepository. We use the *Categorical Datasets* from ADRepository (Pang et al., 2021). For each dataset, we randomly split the full labeled data into 70% training and 30% testing using a stratified split with respect to the anomaly label. After the split, all anomalous samples in the training portion are removed, so that only nominal training samples are used for model training or for constructing the non-parametric reference set. The test set is kept unchanged and contains both normal and anomalous samples. Thus, anomalous samples are never observed during training, and labels are used only for evaluation.

All selected ADRepository datasets are categorical tabular datasets. Each feature is represented as a discrete symbol and encoded as an integer ID. The same underlying categorical information is used by MaskDiff-AD and the tabular baselines, with method-specific encodings applied when necessary.

NLP-ADBench. For the NLP anomaly detection datasets, we follow the official train/test split provided by NLP-ADBench (Li et al., 2025), which uses 70% of the data for training and 30% for testing. As in the tabular experiments, only normal samples in the training split are used for training. The full test split, containing both normal and anomalous text samples, is used for computing anomaly scores and evaluation metrics.

For text preprocessing, we convert each raw text sample into a fixed-length discrete sequence using the GPT-2 tokenizer (Radford et al., 2019). Our model does not use the pretrained GPT-2 language model or its pretrained token embeddings; the tokenizer is used only to map raw text into discrete token IDs.

Specifically, we use the GPT-2 tokenizer and apply right padding to obtain fixed-length token sequences. Since the original GPT-2 tokenizer does not provide dedicated padding or mask tokens, we add them when they are absent:

$$\langle \mathrm{pad} \rangle, \qquad \langle \mathsf{m} \rangle.$$

Let $\mathcal{V}$ denote the resulting tokenizer vocabulary, including the original GPT-2 vocabulary and the newly added special tokens. Each text sample is mapped to a sequence of token IDs in $\{0, \ldots, |\mathcal{V}| - 1\}$.

For the final inputs, we truncate or pad every sequence to a fixed length $L_{\max}$. Thus each text sample is represented as

$$x = (x_1, \ldots, x_{L_{\max}}), \qquad x_\ell \in \{0, \ldots, |\mathcal{V}| - 1\}.$$

Texts longer than $L_{\max}$ tokens are truncated, and texts shorter than $L_{\max}$ tokens are padded on the right with <|pad|>. Together with the token IDs, we store an attention mask

$$a = (a_1, \ldots, a_{L_{\max}}), \qquad a_\ell \in \{0, 1\},$$

where $a_\ell = 1$ indicates a valid token position and $a_\ell = 0$ indicates a padding position. Hence, after tokenization, each sample is represented by

$$(x, a) \in \{0, \ldots, |\mathcal{V}| - 1\}^{L_{\max}} \times \{0, 1\}^{L_{\max}}.$$

The model is trained only on the tokenized normal examples from the training split, while the full tokenized test split is used for evaluation. The same tokenizer and maximum sequence length are used for both training and testing.

B.2 Evaluation Criteria

We evaluate anomaly detection performance using ROC-AUC and PR-AUC. For each test sample $\mathbf{x}_i$, let $y_i \in \{0, 1\}$ denote its ground-truth label, where $y_i = 0$ indicates a normal sample and $y_i = 1$ indicates an anomalous sample. Each method produces an anomaly score $S(\mathbf{x}_i)$, where a larger score indicates stronger evidence of anomaly.

ROC-AUC measures the global ranking quality between anomalous and normal samples. Equivalently, it can be interpreted as the probability that a randomly chosen anomalous sample receives a higher score than a randomly chosen normal sample:

$$\mathrm{AUC}_{\mathrm{ROC}} = \mathbb{P}\big(S(\mathbf{x}^+) > S(\mathbf{x}^-)\big),$$

where $\mathbf{x}^+$ denotes a randomly selected anomalous sample and $\mathbf{x}^-$ denotes a randomly selected normal sample.

PR-AUC summarizes the precision–recall tradeoff obtained by thresholding the anomaly scores. For a threshold γ, samples with $S(\mathbf{x}_i) \geq \gamma$ are predicted as anomalies, and

$$\mathrm{Precision}(\gamma) = \frac{\mathrm{TP}(\gamma)}{\mathrm{TP}(\gamma) + \mathrm{FP}(\gamma)}, \qquad \mathrm{Recall}(\gamma) = \frac{\mathrm{TP}(\gamma)}{\mathrm{TP}(\gamma) + \mathrm{FN}(\gamma)}.$$

We report PR-AUC as average precision, which corresponds to a step-wise area under the precision.

B.3 MaskDiff-AD Hyperparameters

For all variants of MaskDiff-AD, we use fixed hyperparameters across datasets rather than validation-based or per-dataset tuning. We adopt the standard linear absorbing schedule $\alpha_t = 1 - t$, so a probe level τ_ℓ has mask probability $1 - \alpha_{\tau_\ell} = \tau_\ell$. All reconstruction scores are averaged over the specified probe levels and independently generated masked probe views.

The parametric tabular model uses the uniform probe grid $\tau_\ell \in \{0.1, 0.2, \ldots, 0.9\}$ and 16 masked probe views per probe level. Its reconstruction network uses embedding dimension 256, hidden dimension 512, three MLP layers, and dropout 0.1. The model is trained for 20 epochs with AdamW using a batch size of 512, learning rate 10^{-3}, and weight decay 10^{-5}.

The non-parametric tabular model uses the probe grid $\tau_\ell \in \{0.15, 0.30, 0.45, 0.60\}$ and 8 masked probe views per probe level. It has no trainable neural parameters and takes all normal training samples as the reference set. Masked-coordinate reconstruction probabilities are estimated using the visible-coordinate Hamming kernel $K_\lambda(\widetilde{\mathbf{x}}, \mathbf{x}^{(n)}) = \exp\{-\lambda d_{\mathrm{vis}}(\widetilde{\mathbf{x}}, \mathbf{x}^{(n)})\}$ with bandwidth $\lambda = 1$, where d_{vis} denotes the Hamming distance restricted to visible coordinates.

For text experiments, we use the GPT-2 tokenizer only, without using the pretrained GPT-2 language model or pretrained GPT-2 token embeddings. The text model uses the same uniform probe grid as the parametric tabular model, $\tau_\ell \in \{0.1, 0.2, \ldots, 0.9\}$, and 24 masked probe views per probe level. The reconstruction model is an 8-layer Transformer encoder with hidden dimension 512, 16 attention heads, feed-forward dimension 2048, and dropout 0.1. It is trained for 20 epochs with AdamW using learning rate 10^{-3} and weight decay 10^{-5}.

Table 3: Default hyperparameters for tabular baselines. The same settings are used across datasets and seeds.

Method	Input	Main settings
COPOD	ordinal/binned	PyOD default
DeepSVDD	one-hot	100 epochs, batch size 64, learning rate 10^{-3}, hidden dims $100, 50$
DRL	one-hot	200 epochs, batch size 512, learning rate 0.05, LR decay 0.98
DTE-Categorical / InvGamma / Gaussian	one-hot	100 epochs, batch size 128, learning rate 10^{-4}, MLP hidden dims $256, 512, 256$
ECOD	ordinal/binned	PyOD default
GOAD	one-hot	100 epochs, batch size 64, learning rate 10^{-3}, 256 transformations
Hamming-kNN	ordinal/binned	$k = 25$, Hamming distance
ICL	one-hot	100 epochs, batch size 64, learning rate 10^{-3}, hidden dims $100, 50$
IForest	one-hot	300 trees, `max_samples=auto`, `contamination=auto`
MCM	one-hot	200 epochs, batch size 512, learning rate 0.05, LR decay 0.98

B.4 Baselines

We compare against classical anomaly detection baselines, DTE-based variants, and recent tabular or text anomaly detection methods. All baselines are evaluated under the same normal-only setting: each method is fitted only on normal training samples, and anomaly labels are used only for test-time evaluation.

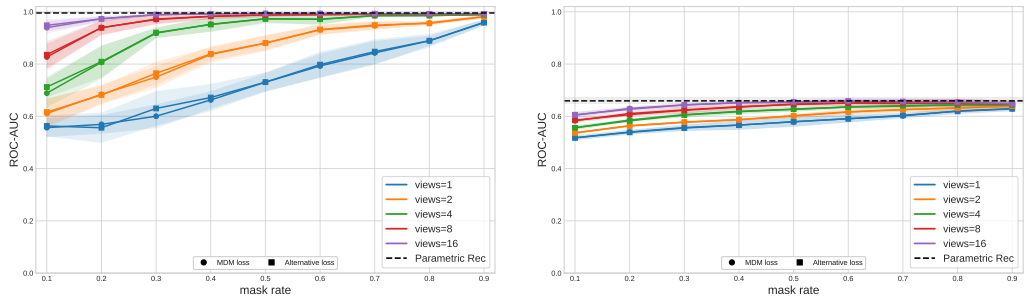
Hyperparameter setup. We use a fixed default hyperparameter configuration across datasets and random seeds, rather than performing per-dataset hyperparameter search. The main settings for all tabular baselines are summarized in Table 3.

Tabular baselines. For categorical and mixed-type tabular datasets, we compare against 12 baseline methods: COPOD (Li et al., 2020), DeepSVDD (Ruff et al., 2018), DTE-Categorical (Livernoche et al., 2024), DTE-InvGamma (Livernoche et al., 2024), DTE-Gaussian (Livernoche et al., 2024), DRL (Ye et al., 2025), ECOD (Li et al., 2023), GOAD (Bergman & Hoshen, 2020), Hamming-kNN (Ramaswamy et al., 2000), ICL (Shenkar & Wolf, 2022), Isolation Forest (IForest) (Liu et al., 2008), and MCM (Yin et al., 2024).

Categorical features are encoded as discrete symbols. Numerical features are discretized by quantile binning using only the normal training subset. By default, we use 10 quantile bins, treat numerical features with at most 20 distinct values as discrete features, and use an explicit unknown bucket for unseen categories. The fitted preprocessing parameters are then reused for the test set. Hamming-kNN, ECOD, and COPOD are applied to the ordinal/binned representation. For methods whose behavior may be affected by artificial ordering of category IDs, including Isolation Forest and neural tabular baselines, we use the one-hot representation. The three DTE variants are also applied to the one-hot representation of the discretized features, ensuring that they use the same underlying discretized information as the other methods.

We additionally compare with three recent tabular anomaly detection methods: NPT-AD (Thimonier et al., 2024), GNSM (Mahmood et al., 2024), and AnoLLM (Tsai et al., 2025). Because these methods have substantially different computational or input constraints, they are evaluated on the subset of datasets for which all compared methods can be run under a controlled setting. The complete results and corresponding computational constraints are reported in Appendix C.1.

Text baselines. For text anomaly detection, we compare against the end-to-end text anomaly detector DATE (Manolache et al., 2021), DTE-based variants (Livernoche et al., 2024), and a broad set of embedding-based anomaly detection methods. For the latter, we use the released BERT (Devlin et al., 2019) and OpenAI text embeddings (OpenAI, 2026) from NLP-ADBench as fixed document representations, followed by LOF (Breunig et al., 2000), DeepSVDD (Ruff et al., 2018), ECOD (Li et al., 2023), Isolation Forest (Liu et al., 2008), SO-GAAL (Liu et al., 2019), AE (Aggarwal, 2017), VAE (Kingma & Welling, 2013), and LUNAR (Goodge et al., 2022). The DTE variants are evaluated on the same released BERT and OpenAI representations. All text baseline results are obtained from our own runs using the same dataset instances, fixed train/test splits, normal-only protocol, and evaluation pipeline as MaskDiff-AD. We report the mean and standard deviation over three random seeds.



(a) U2R (b) Bank

Figure 4: Additional sensitivity analysis of Parametric MaskDiff-AD to the probe mask rate τ on U2R and Bank. Together with the Vehicle Claims result in Figure 3, these results show that the optimal single probe level can vary across datasets, motivating the fixed multi-probe aggregation used in the main experiments.

B.5 Sensitivity Analysis

Figure 4 provides additional sensitivity results on U2R and Bank. Compared with the Vehicle Claims result in Figure 3, these datasets exhibit different best-performing single probe levels. This supports our choice of using a fixed multi-probe score that aggregates over a uniform probe grid, rather than tuning the probe level separately for each dataset. In low-view regimes, the average masked reconstruction loss in Eq. (8) is occasionally marginally better than the original masked diffusion loss on some datasets, possibly because it is more closely aligned with the normalized test-time reconstruction score.

B.6 Runtime and Comparison

We report runtime in Table 4. The comparison is conducted on two tabular datasets with different scales: Bank contains 41,188 samples with 10 features, while Vehicle Claims contains 268,255 samples with 18 features. Runtime is decomposed into training or fitting time, inference time, and total time.

The results show that Parametric MaskDiff-AD achieves a favorable balance between training cost and inference efficiency. On the large Vehicle Claims dataset, Parametric MaskDiff-AD takes 31.64 seconds for training and 16.28 seconds for inference, leading to a total runtime of 47.92 seconds. This is substantially faster than several deep baselines such as GOAD, ICL, and MCM, while also avoiding the high test-time cost of Non-parametric MaskDiff-AD. On the Bank dataset, Parametric MaskDiff-AD remains efficient, with a total runtime of 5.71 seconds.

The comparison also highlights the different computational profiles of the two MaskDiff-AD variants. Non-parametric MaskDiff-AD does not require model training, but its test-time cost can become large because each test sample is compared against the normal training set through masked-view reconstruction. This effect is especially clear on Vehicle Claims, where its inference time reaches 3005.69 seconds. In contrast, Parametric MaskDiff-AD amortizes the reconstruction model during training and then performs inference through neural-network, making it more scalable on larger datasets. For smaller or medium-scale datasets such as Bank, the non-parametric variant is still feasible, but the parametric version provides a more consistent runtime profile across dataset sizes.

C Additional Comparisons

C.1 Comparison with Recent Tabular Anomaly Detection Methods

We additionally compare Parametric MaskDiff-AD with three recent tabular anomaly detection methods: NPT-AD (Thimonier et al., 2024), GNSM (Mahmood et al., 2024), and AnoLLM (Tsai et al., 2025). Because these methods have different computational and input constraints, they were not all successfully evaluated on the full set of fourteen tabular datasets. To enable a controlled

Table 4: Runtime comparison on the Bank and Vehicle Claims datasets.

	Bank			Vehicle Claims		
Method	Train / Fit	Inference	Total	Train / Fit	Inference	Total
Parametric MaskDiff-AD	3.87	1.84	5.71	31.64	16.28	47.92
Non-parametric MaskDiff-AD	0.00	4.81	4.81	0.00	3005.69	3005.69
COPOD	0.01	0.02	0.03	0.13	0.22	0.35
DeepSVDD	18.73	0.03	18.75	130.88	0.20	131.09
DRL	20.24	0.02	20.25	132.68	0.12	132.80
DTE-Categorical	17.21	0.02	17.23	138.35	0.25	138.59
DTE-Gaussian	15.33	0.02	15.35	130.23	0.24	130.47
DTE-InvGamma	18.03	0.03	18.05	144.01	0.25	144.27
ECOD	0.12	0.02	0.14	0.24	0.25	0.49
GOAD	122.88	1.40	124.27	768.66	22.35	791.02
Hamming-kNN	0.00	3.17	3.17	0.00	65.66	65.66
ICL	61.86	0.11	61.97	817.37	2.02	819.39
IForest	0.25	0.09	0.35	0.35	1.35	1.70
MCM	49.69	0.04	49.72	678.01	0.63	678.64

head-to-head comparison, we therefore report results on the six-dataset intersection for which all four methods are available: `bank`, `chess`, `cmc`, `probe`, `solar`, and `u2r`.

Table 5 reports ROC-AUC and PR-AUC on this common subset. In addition to the average score, we report the number of wins/ties/losses (W/T/L) of Parametric MaskDiff-AD against each competing method across the six datasets. This is useful because averages can be influenced by large margins on individual datasets.

Table 5: Comparison with recent tabular anomaly detection methods on the six datasets for which all four methods are available. "Ours W/T/L" denotes the number of datasets on which Parametric MaskDiff-AD obtains a higher, equal, or lower score than the corresponding baseline. Best results for each dataset and the best average result under each metric are shown in bold.

Method	Metric	Average	bank	chess	cmc	probe	solar	u2r	Ours W/T/L
Parametric MaskDiff-AD	ROC-AUC	**80.45**	65.90	**73.55**	**59.30**	**98.42**	86.00	**99.55**	–
NPT-AD	ROC-AUC	76.46	**68.75**	66.13	47.32	97.94	79.14	99.47	5/0/1
GNSM	ROC-AUC	71.22	51.68	46.14	46.64	98.01	85.29	**99.55**	5/1/0
AnoLLM-135M	ROC-AUC	73.59	63.59	47.95	41.25	98.38	**91.28**	99.10	5/0/1
Parametric MaskDiff-AD	PR-AUC	31.45	23.90	**0.69**	**5.89**	**91.59**	20.29	46.37	–
NPT-AD	PR-AUC	27.97	**27.73**	0.19	2.63	76.15	18.91	42.17	5/0/1
GNSM	PR-AUC	**32.88**	11.73	0.13	3.37	89.42	20.21	**72.42**	5/0/1
AnoLLM-135M	PR-AUC	29.42	23.12	0.11	2.00	89.31	**23.24**	38.75	5/0/1

On this common subset, Parametric MaskDiff-AD achieves the highest average ROC-AUC, outperforming NPT-AD, GNSM, and AnoLLM on five of the six datasets in pairwise ROC-AUC comparisons, with one tie against GNSM on `u2r`. For PR-AUC, Parametric MaskDiff-AD also outperforms each competing method on five of the six datasets. GNSM obtains a slightly higher mean PR-AUC (32.88 versus 31.45), primarily due to its substantially higher PR-AUC on `u2r`; the per-dataset win counts therefore provide a complementary view of performance consistency.

These comparisons also highlight practical scalability differences among the methods. NPT-AD relies on feature-wise attention and repeated feature-reconstruction operations, causing its computational and memory costs to increase substantially with dimensionality and dataset size. AnoLLM serializes each tabular row into a token sequence and is therefore constrained by the context length of the underlying language model. In our experiments with its SmolLM-135M backbone, the `ad` and `w7a` datasets could not be evaluated because individual serialized rows exceeded the 2,048-token context window. GNSM does not impose the same explicit dimensionality or context-length limit, but its optimization procedure is substantially more computationally expensive.

In contrast, Parametric MaskDiff-AD was evaluated on all fourteen tabular datasets in our main benchmark, including the substantially higher-dimensional datasets. Thus, in addition to the controlled six-dataset comparison above, the full benchmark illustrates an important practical advantage of the proposed discrete masked-reconstruction formulation in larger and higher-dimensional settings.

C.2 Comparison with DTE on Compact Continuous Embeddings

Our analysis of DTE under masked diffusion concerns its direct application in the native absorbing masked state space and does not imply that DTE degenerates after mapping discrete data to an external continuous representation. To examine whether the sparse one-hot representation used in our main DTE baselines disadvantages DTE, we additionally evaluate DTE on a compact continuous embedding.

Specifically, we first apply the same training-only discretization and one-hot encoding used in the main experiments. We then fit a 32-dimensional PCA projection using only nominal training data and standardize the projected features before applying DTE. All methods use the same data splits, five random seeds, and DTE hyperparameter settings as in the main experiments.

Table 6 reports the average performance across the evaluated datasets. Compressing the one-hot representation to 32 dimensions does not improve DTE: for all three DTE parameterizations, the PCA-32 representation yields lower average ROC-AUC and PR-AUC than the corresponding one-hot representation. The average ROC-AUC of the PCA-based DTE variants ranges from 67.54 to 72.90, compared with 74.38 to 78.86 for their one-hot counterparts. Parametric MaskDiff-AD achieves the highest average ROC-AUC of 80.66. For PR-AUC, DTE-Gaussian with the one-hot representation achieves the highest average value of 35.34, while Parametric MaskDiff-AD achieves 32.39. Overall, these results show that replacing the sparse one-hot representation with a compact PCA embedding does not improve DTE or alter the main comparison.

Table 6: Comparison with DTE using one-hot and compact PCA representations. PCA-32 denotes a 32-dimensional PCA projection fitted using nominal training data only. Best results in each metric are shown in bold.

Method	Representation	Avg. ROC-AUC	Avg. PR-AUC
Parametric MaskDiff-AD	Native discrete	**80.66**	32.39
DTE-Categorical	One-hot	78.86	25.79
DTE-Categorical	PCA-32	72.90	21.12
DTE-InvGamma	One-hot	74.38	29.17
DTE-InvGamma	PCA-32	67.54	19.37
DTE-Gaussian	One-hot	77.29	**35.34**
DTE-Gaussian	PCA-32	70.74	23.26

C.3 Comparison with Baselines on Raw Continuous Features

In the main experiments, continuous variables in mixed-type datasets are discretized into ten bins so that all methods are evaluated using the same underlying representation. We additionally evaluate the baseline methods on the mixed-type Vehicle Claims dataset using their original continuous numerical variables, while keeping the categorical preprocessing unchanged. Parametric MaskDiff-AD continues to use the discretized representation required by its discrete-state formulation.

Table 7 reports the results. Parametric MaskDiff-AD achieves the highest ROC-AUC and PR-AUC, with scores of 96.38 and 93.04, respectively. Its performance remains substantially higher than that of the baselines evaluated with their native continuous numerical inputs. These results indicate that the performance advantage of MaskDiff-AD on the mixed-type dataset is not attributable to forcing continuous-input baselines to operate on discretized numerical features.

C.4 Ablation of Test-Time Scoring Strategies

To examine the effect of test-time scoring strategies, we conduct a matched-checkpoint and matched-compute ablation using the same trained parametric reconstruction model. We compare Parametric MaskDiff-AD with two likelihood-based alternatives. Fixed-grid NELBO uses the same probe grid $\tau_\ell \in \{0.1, \ldots, 0.9\}$ but normalizes by the expected number of masked coordinates rather than the realized $|M(\tilde{\mathbf{x}})|$, while MC per-token NELBO allocates the same total inference budget over randomly sampled corruption times. Thus, the comparison isolates the effects of realized-mask normalization and fixed-grid allocation at test time.

Table 7: Anomaly detection performance on the Vehicle Claims dataset when numerical variables are retained in their original continuous form for baseline methods. Parametric MaskDiff-AD uses the discretized representation required by its discrete-state formulation.

Method	ROC-AUC	PR-AUC
Parametric MaskDiff-AD	**96.38**	**93.04**
COPOD	63.93	33.56
DeepSVDD	58.31	41.28
DRL	87.23	80.00
DTE-Categorical	66.26	35.17
DTE-Gaussian	84.73	72.32
DTE-InvGamma	87.94	82.50
ECOD	63.51	34.11
GOAD	48.04	24.96
ICL	87.77	82.13
Isolation Forest	48.21	21.02
MCM	63.02	41.83

Table 8: Sensitivity of non-parametric MaskDiff-AD to the probe grid. ROC-AUC and PR-AUC are averages over the 14 tabular datasets and are reported in percentage points.

Probe grid	K	Total views	ROC-AUC	PR-AUC
$\{0.15, 0.30, 0.45, 0.60\}$	8	32	79.12	31.64
$\{0.1, 0.2, \ldots, 0.9\}$	4	36	**79.95**	**31.68**

The complete results over fourteen tabular datasets are reported in Tables 13 and 14. Parametric MaskDiff-AD achieves average ROC-AUC/PR-AUC of $80.66/32.39$, compared with $79.97/32.35$ for Fixed-grid NELBO and $79.88/32.18$ for MC per-token NELBO. Against Fixed-grid NELBO, MaskDiff-AD records $9/2/3$ wins/ties/losses in ROC-AUC and $8/2/4$ in PR-AUC; against MC per-token NELBO, the corresponding counts are $6/2/6$ and $6/1/7$. The largest ROC-AUC difference occurs on `chess`, while the three scoring rules perform similarly on most other datasets.

Overall, the results suggest that the masked-conditional likelihood signal provides most of the anomaly-detection performance, while realized-mask normalization and fixed-grid evaluation provide modest and dataset-dependent refinements to the test-time score.

C.5 Sensitivity of the Non-Parametric Variant

We further examine the sensitivity of the non-parametric MaskDiff-AD to two hyperparameter choices: the probe grid and the kernel bandwidth λ in Eq. (7). Unless otherwise specified, the main non-parametric configuration uses the four-level probe grid $\{0.15, 0.30, 0.45, 0.60\}$, $K = 8$ masked views per level, and $\lambda = 1$, as described in Appendix B.3. We report average ROC-AUC and PR-AUC over all 14 tabular datasets.

Sensitivity to the probe grid. To evaluate whether the non-parametric method depends strongly on probe grids, we rerun it using the same nine-level uniform grid $\{0.1, 0.2, \ldots, 0.9\}$ used by the parametric variant. To approximately match the original inference budget, we use $K = 4$ views per level, resulting in 36 total masked views, compared with 32 views in the default configuration. Both configurations use $\lambda = 1$.

As shown in Table 8, replacing the four-level grid with the substantially denser nine-level grid changes the average ROC-AUC by only 0.83 percentage points and the average PR-AUC by only 0.04 percentage points. The similar aggregate performance under two substantially different grids indicates that the non-parametric method is not highly sensitive to the precise choice of probe grid.

Sensitivity to the kernel bandwidth λ. We next vary the bandwidth parameter in the visible-coordinate Hamming kernel

$$K_\lambda(\widetilde{\mathbf{x}}, \mathbf{x}^{(n)}) = \exp\Big\{-\lambda d_{\text{vis}}(\widetilde{\mathbf{x}}, \mathbf{x}^{(n)})\Big\},$$

Table 9: Sensitivity of non-parametric MaskDiff-AD to the kernel bandwidth λ. ROC-AUC and PR-AUC are averages over the 14 tabular datasets and are reported in percentage points.

λ	ROC-AUC	PR-AUC
1	79.12	31.64
2	**79.53**	**31.87**
3	78.93	31.54

while keeping the default non-parametric probe configuration fixed. Table 9 reports the resulting average performance for $\lambda \in \{1, 2, 3\}$.

Performance remains stable across the tested bandwidths. The range across $\lambda \in \{1, 2, 3\}$ is only 0.60 percentage points in ROC-AUC and 0.33 percentage points in PR-AUC. Although $\lambda = 2$ gives the highest average performance, the differences are small, indicating that non-parametric MaskDiff-AD is not highly sensitive to the kernel bandwidth within this range. Together with the probe-grid experiment, these results suggest that the non-parametric variant is robust to reasonable changes in its main test-time hyperparameters.

D Proofs and Discussions of Theoretical Guarantee

Proof of Theorem 1. We condition throughout on the trained reconstruction model $p_{\hat{\theta}}$. All probabilities below are taken jointly over the test sample and the randomly generated probe views, unless we explicitly condition on the test sample. For each probe level τ_ℓ and view k, define

$$Z_{\ell,k}(\mathbf{x}) := s_{\mathrm{rec}}\big(\mathbf{x}; \widetilde{\mathbf{x}}^{(\ell,k)}\big), \qquad \widetilde{\mathbf{x}}^{(\ell,k)} \sim q_{\tau_\ell|0}(\cdot \mid \mathbf{x}).$$

Conditional on $\mathbf{x}$, the random variables $\{Z_{\ell,k}(\mathbf{x})\}_{\ell\in[L],k\in[K]}$ are independent, although they are not necessarily identically distributed across the probe levels. By the assumptions in Theorem 1, we have that $|Z_{\ell,k}(\mathbf{x})| \le C_\ell$ for all $\mathbf{x}$, ℓ, and k. By the definitions of S_{rec} and S^*_{rec}, we have

$$S_{\mathrm{rec}}(\mathbf{x}) = \frac{1}{LK}\sum_{\ell=1}^{L}\sum_{k=1}^{K} Z_{\ell,k}(\mathbf{x}),$$
$$S^*_{\mathrm{rec}}(\mathbf{x}) = \frac{1}{LK}\sum_{\ell=1}^{L}\sum_{k=1}^{K} \mathbb{E}\left[Z_{\ell,k}(\mathbf{x}) \mid \mathbf{x}\right].$$

Thus, for any $t > 0$, Hoeffding's inequality for independent, but not necessarily identically distributed, bounded random variables gives

$$\mathbb{P}\left(S_{\mathrm{rec}}(\mathbf{x}) - S^*_{\mathrm{rec}}(\mathbf{x}) \ge t \mid \mathbf{x}\right) \le \exp\left(-\frac{2L^2Kt^2}{\sum_{\ell=1}^{L} C_\ell^2}\right), \tag{11}$$

$$\mathbb{P}\left(S^*_{\mathrm{rec}}(\mathbf{x}) - S_{\mathrm{rec}}(\mathbf{x}) \ge t \mid \mathbf{x}\right) \le \exp\left(-\frac{2L^2Kt^2}{\sum_{\ell=1}^{L} C_\ell^2}\right). \tag{12}$$

Since the right-hand sides of Eqs. (11)–(12) do not depend on $\mathbf{x}$, the same bounds hold after averaging over either $\mathbf{x} \sim q_0$ or $\mathbf{x} \sim q'$.

Furthermore, from the definitions for S^* and S^*_{rec} in Eqs. (9) and (10), respectively, we also have

$$\Delta_{D_{\mathrm{KL}}}(\mathbf{x}) = S^*_{\mathrm{rec}}(\mathbf{x}) - S^*(\mathbf{x}) = \frac{1}{L}\sum_{\ell=1}^{L} \mathbb{E}_{\widetilde{\mathbf{x}}\sim q_{\tau_\ell|0}(\cdot|\mathbf{x})} \frac{1}{|M(\widetilde{\mathbf{x}})| \vee 1}\Big(\sum_{j\in M(\widetilde{\mathbf{x}})} \log \frac{q^j\big(x_j \mid \widetilde{\mathbf{x}}, \tau_\ell\big)}{p^j_{\hat{\theta}}\big(x_j \mid \widetilde{\mathbf{x}}, \tau_\ell\big)}\Big). \tag{13}$$

Since the level-ℓ reconstruction score is bounded by C_ℓ, we have

$$|S_{\mathrm{rec}}(\mathbf{x})| \le \overline{C}, \quad |S^*_{\mathrm{rec}}(\mathbf{x})| \le \overline{C}, \quad |S^*(\mathbf{x})| \le \overline{C}, \quad |\Delta_{D_{\mathrm{KL}}}(\mathbf{x})| \le 2\overline{C}.$$

In the following, we provide the proofs for the Type-I error and the Type-II error, respectively.

Type-I error Under the normal-side approximation condition $\mathbb{E}_{\mathbf{x}\sim q_0}\left[\Delta_{D_{\mathrm{KL}}}(\mathbf{x})\right] \le \epsilon_0$, we have

$$\overline{\mu}_0 := \mathbb{E}_{\mathbf{x}\sim q_0}\left[S^*_{\mathrm{rec}}\right] = \mu^*_0 + \mathbb{E}_{\mathbf{x}\sim q_0}\left[\Delta_{D_{\mathrm{KL}}}(\mathbf{x})\right] \le \mu^*_0 + \epsilon_0. \tag{14}$$

The bounded-variable form of Hoeffding's inequality gives, for any $u > 0$,

$$\mathbb{P}_{\mathbf{x}\sim q_0}\left(S^*_{\mathrm{rec}}(\mathbf{x}) - \overline{\mu}_0 > u\right) \le \exp\left(-\frac{2u^2}{\overline{C}^2}\right). \tag{15}$$

For simplicity, we define $\delta_0 := \gamma - \mu^*_0 - \epsilon_0$ and suppose that $\delta_0 > 0$. Taking $t = \delta_0/2$, it follows from Eq. (14) that $\gamma - t - \overline{\mu}_0 \ge \frac{\delta_0}{2}$. Combining Eq. (11) with Eq. (15), we obtain

$$\begin{aligned}\mathbb{P}_{\mathbf{x}\sim q_0}\left(S_{\mathrm{rec}}(\mathbf{x}) > \gamma\right) &\le \mathbb{P}_{\mathbf{x}\sim q_0}\left(S_{\mathrm{rec}}(\mathbf{x}) - S^*_{\mathrm{rec}}(\mathbf{x}) > \frac{\delta_0}{2}\right) + \mathbb{P}_{\mathbf{x}\sim q_0}\left(S^*_{\mathrm{rec}}(\mathbf{x}) - \overline{\mu}_0 > \frac{\delta_0}{2}\right) \\ &\le \exp\left(-\frac{L^2 K \delta_0^2}{2\sum_{\ell=1}^{L} C_\ell^2}\right) + \exp\left(-\frac{\delta_0^2}{2\overline{C}^2}\right).\end{aligned}$$

Type-II error For the anomalous distribution q', by the definition of the one-sided misspecification error ϵ_1, we have

$$\overline{\mu}_1 := \mathbb{E}_{\mathbf{x}\sim q'}\left[S^*_{\mathrm{rec}}\right] \ge \mu^*_1 - \epsilon_1, \tag{16}$$

and

$$\mathbb{P}_{\mathbf{x}\sim q'}\left(\overline{\mu}_1 - S^*_{\mathrm{rec}}(\mathbf{x}) > u\right) \le \exp\left(-\frac{2u^2}{\overline{C}^2}\right). \tag{17}$$

Define $\delta_1 := \mu^*_1 - \epsilon_1 - \gamma$ and suppose that $\delta_1 > 0$. Taking $t = \delta_1/2$, it follows from Eq. (16) that $\overline{\mu}_1 - \gamma - t \ge \frac{\delta_1}{2}$. Combining Eq. (12) with Eq. (17), we obtain

$$\begin{aligned}\mathbb{P}_{\mathbf{x}\sim q'}\left(S_{\mathrm{rec}}(\mathbf{x}) \le \gamma\right) &\le \mathbb{P}_{\mathbf{x}\sim q'}\left(S^*_{\mathrm{rec}}(\mathbf{x}) - S_{\mathrm{rec}}(\mathbf{x}) \ge \frac{\delta_1}{2}\right) + \mathbb{P}_{\mathbf{x}\sim q'}\left(\overline{\mu}_1 - S^*_{\mathrm{rec}}(\mathbf{x}) \ge \frac{\delta_1}{2}\right) \\ &\le \exp\left(-\frac{L^2 K \delta_1^2}{2\sum_{\ell=1}^{L} C_\ell^2}\right) + \exp\left(-\frac{\delta_1^2}{2\overline{C}^2}\right).\end{aligned}$$

In particular, when $C_\ell = C$ for every ℓ, we have $\overline{C} = C$ and $\sum_{\ell=1}^{L} C_\ell^2 = LC^2$. Hence, the preceding bounds simplify to

$$\begin{aligned}\mathbb{P}_{\mathbf{x}\sim q_0}\left(S_{\mathrm{rec}}(\mathbf{x}) > \gamma\right) &\le \exp\left(-\frac{LK(\gamma - \mu^*_0 - \epsilon_0)^2}{2C^2}\right) + \exp\left(-\frac{(\gamma - \mu^*_0 - \epsilon_0)^2}{2C^2}\right), \\ \mathbb{P}_{\mathbf{x}\sim q'}\left(S_{\mathrm{rec}}(\mathbf{x}) \le \gamma\right) &\le \exp\left(-\frac{LK(\mu^*_1 - \epsilon_1 - \gamma)^2}{2C^2}\right) + \exp\left(-\frac{(\mu^*_1 - \epsilon_1 - \gamma)^2}{2C^2}\right).\end{aligned}$$

□

Remark 3 (Model misspecificaiton error)**.** *The quantity ϵ_1 defined in Theorem 1 measures the extent to which model misspecification makes anomalous samples easier to reconstruct than under the oracle nominal conditionals. One example of a sufficient condition to guarantee a small ϵ_1 is as follows. Suppose that for every probe level τ_ℓ, masked context in the support of q', masked coordinate j, and categorical value a, there exists a small η_ℓ such that*

$$p^j_{\hat{\theta}}(a \mid \widetilde{\mathbf{x}}, \tau_\ell) \le e^{\eta_\ell} q^j(a \mid \widetilde{\mathbf{x}}, \tau_\ell).$$

Then,

$$\epsilon_1 \le \frac{1}{L}\sum_{\ell=1}^{L} \eta_\ell.$$

In other words, ϵ_1 is small whenever the learned reconstruction model does not substantially overestimate the oracle nominal likelihood on the masked configurations generated under q'. For example, this could happen when the learned model is accurate regardless of the underlying measure.

For completeness, below we also provide the detection performance guarantee under the idealized setting where $\epsilon_0 = \epsilon_1 = 0$, i.e., the model used to compute the reconstruction score coincides exactly with the oracle model.

Theorem 2 (Idealized detection performance). *For each probe level τ_ℓ, assume there exists $C_\ell > 0$ such that*

$$-\log p^j_{\widehat{\theta}_{\mathrm{rec}}}(x_j \mid \widetilde{\mathbf{x}}, \tau_\ell) \in [0, C_\ell]$$

for all j, $\mathbf{x}$, and masked views $\widetilde{\mathbf{x}}$. Assume that the LK probe views are sampled independently conditional on $\mathbf{x}$, and define $C^2_{\mathrm{rms}} := \frac{1}{L}\sum_{\ell=1}^{L} C^2_\ell$. Let

$$F_0(u) := \mathbb{P}_{\mathbf{x}\sim q_0}\left(S^*_{\mathrm{rec}}(\mathbf{x}) \le u\right), \qquad F_1(u) := \mathbb{P}_{\mathbf{x}\sim q'}\left(S^*_{\mathrm{rec}}(\mathbf{x}) \le u\right)$$

be the distribution functions of the population score under the normal and anomalous distributions, respectively. Then, for any fixed threshold γ and any $t > 0$, we have

- *Type-I error:*

$$\mathbb{P}_{\mathbf{x}\sim q_0}\left(S_{\mathrm{rec}}(\mathbf{x}) > \gamma\right) \le \exp\left(-\frac{2LKt^2}{C^2_{\mathrm{rms}}}\right) + 1 - F_0(\gamma - t).$$

- *Type-II error:*

$$\mathbb{P}_{\mathbf{x}\sim q'}\left(S_{\mathrm{rec}}(\mathbf{x}) \le \gamma\right) \le \exp\left(-\frac{2LKt^2}{C^2_{\mathrm{rms}}}\right) + F_1(\gamma + t).$$

The probabilities are taken jointly over the test sample and the randomly sampled probe views.

In particular, for any target Type-I error level $\alpha \in (0, 1)$, define

$$t_\alpha := C_{\mathrm{rms}}\sqrt{\frac{\log(2/\alpha)}{2LK}}$$

and

$$\gamma_\alpha := F_0^{-1}(1 - \alpha/2) + t_\alpha,$$

where F_0^{-1} denotes the generalized inverse of F_0. Then

$$\mathbb{P}_{\mathbf{x}\sim q_0}\left(S_{\mathrm{rec}}(\mathbf{x}) > \gamma_\alpha\right) \le \alpha.$$

Proof of Theorem 2. We follow the notation of Theorem 1.

Type-I error For any $t > 0$, note that

$$\{S_{\mathrm{rec}}(\mathbf{x}) > \gamma\} \subseteq \{S_{\mathrm{rec}}(\mathbf{x}) - S^*_{\mathrm{rec}}(\mathbf{x}) > t\} \cup \{S^*_{\mathrm{rec}}(\mathbf{x}) > \gamma - t\}.$$

So, we have

$$\begin{aligned}\mathbb{P}_{\mathbf{x}\sim q_0}\left(S_{\mathrm{rec}}(\mathbf{x}) > \gamma\right) &\overset{(i)}{\le} \mathbb{P}_{\mathbf{x}\sim q_0}\left(S_{\mathrm{rec}}(\mathbf{x}) - S^*_{\mathrm{rec}}(\mathbf{x}) > t\right) + \mathbb{P}_{\mathbf{x}\sim q_0}\left(S^*_{\mathrm{rec}}(\mathbf{x}) > \gamma - t\right) \\ &\overset{(ii)}{\le} \exp\left(-\frac{2L^2Kt^2}{\sum_{\ell=1}^{L} C^2_\ell}\right) + 1 - F_0(\gamma - t),\end{aligned}$$

where (i) follows from the union bound, and (ii) follows from Eq. (11) and the definition of F_0.

Type-II error The analysis is similar. For any $t > 0$, note that

$$\{S_{\mathrm{rec}}(\mathbf{x}) \le \gamma\} \subseteq \{S^*_{\mathrm{rec}}(\mathbf{x}) - S_{\mathrm{rec}}(\mathbf{x}) > t\} \cup \{S^*_{\mathrm{rec}}(\mathbf{x}) \le \gamma + t\}.$$

Therefore,

$$\begin{aligned}\mathbb{P}_{\mathbf{x}\sim q'}\left(S_{\mathrm{rec}}(\mathbf{x}) \le \gamma\right) &\overset{(iii)}{\le} \mathbb{P}_{\mathbf{x}\sim q'}\left(S^*_{\mathrm{rec}}(\mathbf{x}) - S_{\mathrm{rec}}(\mathbf{x}) > t\right) + \mathbb{P}_{\mathbf{x}\sim q'}\left(S^*_{\mathrm{rec}}(\mathbf{x}) \le \gamma + t\right) \\ &\overset{(iv)}{\le} \exp\left(-\frac{2L^2Kt^2}{\sum_{\ell=1}^{L} C^2_\ell}\right) + F_1(\gamma + t),\end{aligned}$$

where (iii) follows again from the union bound, and (iv) follows from Eq. (12) and the definition of F_1.

□

E COMPLETE EXPERIMENTAL RESULTS

Table 10: ROC-AUC on tabular datasets over five random seeds.

Method	ad	aid362	apas	bank	census	chess	cmc	covertype	probe	r10	solar	u2r	vehicle_claims	w7a
Parametric MaskDiff-AD	78.86 (2.34)	67.93 (4.78)	83.15 (3.36)	65.90 (0.91)	66.62 (0.52)	73.55 (7.74)	59.30 (8.97)	91.92 (0.68)	98.42 (0.09)	99.14 (0.14)	86.00 (4.61)	99.55 (0.10)	**96.38 (0.09)**	62.53 (1.93)
Non-parametric MaskDiff-AD	**93.43 (1.72)**	64.69 (4.70)	80.01 (1.85)	63.68 (0.89)	67.01 (0.15)	55.96 (11.17)	56.85 (11.64)	95.47 (0.21)	98.38 (0.09)	99.09 (0.12)	85.85 (3.48)	99.11 (0.12)	83.14 (0.15)	65.05 (1.44)
COPOD	73.85 (2.08)	65.12 (4.77)	64.15 (3.96)	59.31 (2.19)	64.45 (1.25)	70.76 (27.06)	51.07 (10.94)	97.72 (0.08)	98.33 (0.08)	98.83 (0.13)	87.04 (5.04)	97.74 (0.67)	67.18 (0.27)	58.68 (1.49)
DeepSVDD	74.09 (12.18)	59.93 (12.83)	57.45 (17.30)	52.05 (2.73)	58.57 (7.34)	45.68 (10.43)	62.17 (5.69)	47.84 (29.21)	83.32 (4.83)	96.88 (3.31)	81.76 (3.32)	76.77 (24.13)	77.41 (1.46)	81.23 (3.32)
DTE-Categorical	85.18 (2.56)	66.53 (3.26)	86.37 (2.16)	**68.05 (1.06)**	67.17 (0.75)	51.97 (16.57)	**63.26 (6.77)**	94.21 (1.06)	97.88 (0.26)	98.91 (0.16)	78.06 (9.51)	99.01 (0.15)	65.82 (0.30)	81.56 (1.10)
DTE-InvGamma	79.30 (6.47)	58.86 (11.71)	75.50 (6.23)	58.19 (5.85)	61.98 (4.16)	57.41 (7.37)	52.98 (13.21)	88.38 (8.25)	91.65 (5.73)	90.30 (4.41)	72.58 (7.86)	85.17 (29.79)	87.70 (0.84)	81.39 (1.54)
DTE-Gaussian	91.27 (1.19)	66.57 (5.09)	72.40 (2.83)	62.52 (0.90)	61.34 (3.37)	57.71 (12.17)	56.48 (8.72)	64.68 (15.46)	98.45 (0.10)	**99.32 (0.11)**	82.84 (4.87)	**99.61 (0.07)**	84.65 (0.54)	**84.13 (2.55)**
DRL	79.95 (1.49)	65.59 (6.76)	73.87 (6.00)	58.72 (2.08)	58.62 (8.07)	45.64 (13.10)	61.60 (8.43)	44.53 (16.41)	**98.88 (0.24)**	99.09 (0.48)	82.13 (5.89)	97.04 (3.18)	68.31 (5.31)	64.01 (12.23)
ECOD	73.92 (2.08)	65.59 (4.63)	67.21 (3.51)	60.90 (1.67)	62.89 (2.28)	62.21 (25.77)	54.13 (9.53)	**97.82 (0.07)**	98.35 (0.10)	98.83 (0.13)	85.78 (4.44)	97.53 (1.38)	66.86 (0.51)	59.13 (1.48)
GOAD	74.72 (2.42)	65.00 (4.48)	65.10 (4.91)	57.06 (1.25)	60.36 (1.24)	62.28 (12.66)	53.27 (9.48)	95.43 (0.82)	98.39 (0.10)	98.72 (0.14)	**87.06 (3.97)**	98.96 (0.12)	58.04 (0.96)	50.03 (1.84)
Hamming-kNN	80.97 (2.02)	**71.69 (3.49)**	**86.89 (1.90)**	64.50 (0.77)	66.40 (0.27)	**93.22 (1.87)**	60.59 (9.43)	50.00 (0.00)	89.17 (0.77)	98.90 (0.12)	83.67 (5.76)	87.87 (1.53)	72.98 (0.19)	52.36 (1.35)
ICL	79.69 (4.19)	60.49 (3.52)	44.57 (18.25)	60.81 (2.53)	65.53 (1.94)	57.56 (7.41)	57.17 (18.61)	36.11 (21.32)	23.85 (7.33)	98.70 (0.27)	82.33 (9.17)	28.50 (31.30)	81.43 (2.65)	76.58 (2.93)
IForest	42.17 (1.58)	66.49 (4.75)	52.75 (1.09)	57.49 (0.94)	59.32 (1.80)	68.19 (14.57)	56.49 (9.64)	93.83 (2.45)	96.67 (0.26)	96.90 (0.51)	83.16 (5.45)	96.69 (0.85)	46.15 (1.71)	41.26 (1.69)
MCM	75.47 (5.24)	63.45 (5.47)	67.51 (3.48)	60.73 (1.94)	**68.18 (0.40)**	56.73 (22.41)	55.83 (12.21)	91.43 (5.43)	95.78 (5.30)	98.61 (0.20)	64.54 (19.06)	95.65 (5.55)	69.75 (2.51)	54.05 (5.72)

Table 11: PR-AUC on tabular datasets over five random seeds.

Method	ad	aid362	apas	bank	census	chess	cmc	covertype	probe	r10	solar	u2r	vehicle_claims	w7a
Parametric MaskDiff-AD	64.40 (3.96)	3.76 (0.94)	4.91 (1.17)	23.90 (1.19)	9.06 (0.09)	0.69 (0.48)	5.89 (3.37)	5.50 (0.83)	91.59 (0.68)	**70.34 (3.75)**	20.29 (6.55)	46.37 (6.45)	**93.04 (0.06)**	13.71 (2.47)
Non-parametric MaskDiff-AD	**86.82 (3.30)**	4.02 (0.88)	5.37 (1.14)	21.39 (0.83)	9.05 (0.11)	0.29 (0.30)	5.45 (4.98)	9.69 (0.40)	89.88 (0.34)	69.03 (3.78)	17.48 (4.24)	34.60 (4.06)	74.39 (0.28)	15.54 (2.56)
COPOD	56.18 (4.18)	2.83 (0.37)	2.46 (0.44)	17.42 (2.41)	8.34 (0.32)	0.68 (0.82)	3.95 (2.19)	12.84 (0.47)	82.92 (0.67)	60.20 (4.51)	21.89 (6.23)	35.63 (12.34)	38.89 (0.42)	4.66 (0.40)
DeepSVDD	52.23 (17.63)	2.61 (1.23)	2.63 (1.67)	13.84 (0.83)	8.74 (2.10)	0.10 (0.03)	5.11 (1.89)	1.57 (1.12)	74.61 (4.29)	68.71 (5.30)	22.14 (5.65)	33.54 (18.24)	71.22 (1.15)	**55.84 (7.10)**
DTE-Categorical	59.87 (7.51)	3.24 (0.28)	5.11 (1.09)	**27.71 (1.74)**	9.12 (0.21)	0.15 (0.10)	3.83 (0.61)	4.98 (0.25)	73.54 (2.60)	52.03 (4.20)	**22.90 (10.74)**	22.83 (2.66)	34.93 (0.27)	40.88 (4.20)
DTE-InvGamma	55.73 (15.85)	4.50 (2.79)	5.56 (2.33)	18.83 (3.44)	8.11 (0.74)	0.14 (0.04)	3.19 (0.67)	6.04 (2.39)	79.05 (6.78)	30.92 (9.94)	19.15 (8.40)	41.56 (29.44)	82.69 (1.46)	52.83 (3.56)
DTE-Gaussian	81.42 (2.97)	**5.07 (2.18)**	4.38 (1.07)	19.81 (1.39)	8.30 (0.42)	0.18 (0.07)	3.68 (1.61)	2.51 (0.90)	91.86 (0.52)	66.04 (5.54)	21.82 (8.78)	**71.36 (3.45)**	72.39 (0.93)	45.98 (3.46)
DRL	53.25 (5.60)	3.58 (1.22)	3.24 (1.23)	16.11 (1.73)	7.23 (1.41)	0.10 (0.05)	6.42 (3.07)	1.36 (1.92)	**92.07 (0.57)**	66.69 (12.28)	21.93 (7.59)	57.29 (15.54)	53.97 (8.83)	24.22 (14.89)
ECOD	56.29 (4.17)	2.85 (0.37)	2.63 (0.42)	19.16 (2.18)	8.01 (0.54)	0.28 (0.21)	3.82 (1.89)	**13.00 (0.47)**	83.03 (1.61)	60.22 (4.51)	21.85 (5.43)	33.75 (15.29)	38.53 (0.61)	4.77 (0.41)
GOAD	54.98 (4.92)	2.72 (0.35)	2.62 (0.68)	15.40 (1.06)	7.27 (0.23)	0.30 (0.28)	**6.98 (8.48)**	10.21 (2.13)	90.09 (1.05)	56.83 (3.89)	21.23 (6.15)	47.75 (12.90)	26.84 (0.68)	2.96 (0.17)
Hamming-kNN	63.64 (3.75)	3.48 (0.57)	**6.91 (1.09)**	21.16 (0.88)	8.76 (0.10)	**1.86 (1.34)**	5.99 (5.52)	0.47 (0.00)	78.90 (1.40)	59.61 (4.34)	17.87 (5.56)	60.29 (4.39)	46.22 (0.32)	3.07 (0.11)
ICL	58.13 (6.48)	3.32 (0.80)	1.33 (0.50)	20.19 (3.51)	8.54 (0.60)	0.47 (0.58)	4.26 (3.15)	1.05 (0.65)	21.48 (7.54)	61.78 (3.56)	19.85 (8.81)	22.91 (30.33)	74.25 (2.89)	33.82 (3.66)
IForest	11.16 (0.27)	3.02 (0.56)	1.58 (0.12)	15.23 (0.49)	7.03 (0.33)	0.50 (0.67)	4.75 (3.37)	5.93 (2.15)	67.72 (2.57)	39.91 (5.05)	18.50 (5.90)	19.14 (1.50)	19.47 (0.75)	2.32 (0.08)
MCM	47.72 (9.38)	2.63 (0.61)	2.31 (0.53)	19.20 (2.98)	**10.14 (0.24)**	1.80 (3.27)	3.85 (1.48)	6.83 (2.34)	85.97 (4.36)	56.23 (7.23)	14.25 (11.54)	42.34 (22.82)	41.25 (3.46)	7.16 (2.24)

Table 12: Anomaly detection results on NLP datasets over three random seeds.

Method	AGNews		Email Spam		SMS Spam		YelpReview	
	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC
Parametric MaskDiff-AD	76.03 (0.02)	31.11 (0.08)	**97.55 (0.09)**	**88.87 (0.19)**	**95.69 (0.07)**	**72.15 (0.24)**	58.82 (0.68)	20.42 (0.33)
BERT + DTE-Categorical	67.92 (6.24)	22.34 (6.36)	74.98 (6.47)	35.55 (5.75)	57.76 (4.94)	13.64 (2.44)	63.02 (1.27)	23.53 (1.21)
BERT + DTE-InvGamma	58.23 (10.08)	16.32 (5.21)	54.43 (13.22)	16.77 (7.95)	34.55 (19.57)	8.63 (3.76)	56.59 (4.11)	19.81 (1.88)
BERT + DTE-Gaussian	64.90 (2.67)	21.07 (1.48)	75.62 (0.87)	34.32 (1.93)	66.55 (3.46)	17.68 (1.93)	56.70 (2.52)	20.48 (1.30)
OpenAI + DTE-Categorical	64.00 (0.12)	18.25 (0.73)	82.93 (1.13)	39.01 (4.98)	72.53 (8.66)	25.39 (14.83)	61.95 (4.51)	23.22 (3.36)
OpenAI + DTE-InvGamma	60.22 (4.06)	18.59 (2.78)	61.91 (16.78)	18.93 (7.79)	54.36 (17.46)	13.06 (7.74)	56.32 (9.98)	22.57 (7.99)
OpenAI + DTE-Gaussian	69.61 (2.21)	26.54 (2.10)	46.16 (3.62)	12.70 (1.59)	58.86 (1.36)	12.35 (0.64)	62.90 (3.22)	25.51 (3.11)
BERT + AE	72.49 (0.60)	22.79 (0.61)	76.30 (0.30)	29.38 (0.07)	69.21 (0.99)	19.32 (0.82)	64.45 (0.13)	25.36 (0.14)
BERT + DeepSVDD	53.26 (5.52)	12.53 (1.84)	66.39 (8.68)	21.23 (4.79)	58.97 (10.19)	14.30 (5.66)	57.83 (3.69)	20.77 (1.63)
BERT + ECOD	63.18 (0.00)	16.16 (0.00)	70.52 (0.00)	20.77 (0.00)	56.06 (0.00)	11.56 (0.00)	59.84 (0.00)	21.97 (0.00)
BERT + iForest	63.17 (2.93)	16.41 (1.52)	70.87 (0.75)	20.71 (0.66)	57.62 (2.77)	12.50 (1.09)	59.22 (0.38)	21.66 (0.17)
BERT + LOF	74.32 (0.00)	25.49 (0.00)	74.82 (0.00)	23.70 (0.00)	71.90 (0.00)	18.37 (0.00)	65.73 (0.00)	26.29 (0.00)
BERT + LUNAR	76.86 (0.30)	26.92 (0.32)	80.30 (0.95)	30.34 (1.05)	70.04 (0.83)	18.08 (0.25)	65.24 (0.03)	26.10 (0.02)
BERT + SO-GAAL	49.09 (3.74)	11.70 (1.40)	50.10 (5.44)	12.58 (1.61)	40.63 (6.40)	8.09 (0.82)	63.15 (0.62)	24.03 (0.43)
BERT + VAE	71.21 (0.99)	21.51 (0.62)	72.76 (2.99)	22.87 (2.57)	59.15 (2.97)	12.71 (1.13)	62.54 (0.27)	24.34 (0.22)
OpenAI + AE	82.71 (0.32)	39.56 (0.77)	92.84 (0.14)	56.28 (0.56)	55.05 (0.50)	10.32 (0.15)	85.36 (6.54)	70.82 (9.03)
OpenAI + DeepSVDD	52.82 (2.36)	12.46 (0.75)	53.33 (16.85)	15.37 (6.01)	48.01 (7.34)	9.84 (1.88)	52.86 (2.37)	18.08 (1.03)
OpenAI + ECOD	76.38 (0.00)	32.94 (0.00)	92.63 (0.00)	55.97 (0.00)	43.17 (0.00)	8.21 (0.00)	**96.88 (0.00)**	**86.39 (0.00)**
OpenAI + iForest	55.10 (5.90)	14.00 (2.46)	77.39 (1.00)	29.33 (0.97)	49.73 (4.56)	10.08 (1.17)	59.25 (4.13)	21.93 (3.05)
OpenAI + LOF	**89.05 (0.00)**	**54.43 (0.00)**	93.92 (0.00)	59.67 (0.00)	78.62 (0.00)	24.50 (0.00)	87.33 (0.00)	57.10 (0.00)
OpenAI + LUNAR	86.35 (3.14)	44.24 (7.73)	93.37 (0.04)	57.71 (0.22)	72.80 (1.33)	16.98 (0.88)	87.21 (0.89)	55.91 (1.03)
OpenAI + SO-GAAL	55.13 (8.12)	13.84 (2.81)	69.61 (22.02)	28.71 (15.80)	29.50 (23.72)	7.90 (3.66)	67.50 (3.71)	27.68 (4.42)
OpenAI + VAE	62.06 (17.54)	18.81 (8.71)	92.80 (0.00)	56.04 (0.01)	40.85 (1.65)	7.89 (0.23)	94.85 (2.85)	84.27 (5.51)
DATE	80.07 (0.37)	37.59 (0.48)	96.89 (0.19)	87.29 (1.18)	94.05 (0.43)	61.86 (1.76)	47.10 (0.18)	15.69 (0.08)

Table 13: ROC-AUC comparison between Parametric MaskDiff-AD and matched NELBO scoring variants over five random seeds.

Method	Avg.	ad	aid362	apas	bank	census	chess	cmc	covertype	probe	r10	solar	u2r	vehicle	w7a
Parametric MaskDiff-AD	**80.66**	78.86 (2.34)	67.93 (4.78)	83.15 (3.36)	**65.90 (0.91)**	66.62 (0.52)	**73.55 (7.74)**	**59.30 (8.97)**	91.92 (0.68)	98.42 (0.09)	**99.14 (0.14)**	86.00 (4.61)	**99.55 (0.10)**	**96.38 (0.09)**	62.53 (1.93)
Fixed-grid NELBO	79.97	78.87 (2.34)	67.87 (4.75)	83.05 (3.63)	65.82 (0.89)	**66.65 (0.52)**	64.64 (5.82)	59.01 (9.58)	**91.96 (0.70)**	98.42 (0.09)	**99.14 (0.14)**	85.80 (4.85)	99.52 (0.11)	96.25 (0.10)	62.52 (1.93)
MC per-token NELBO	79.88	**78.92 (2.32)**	**68.04 (4.71)**	**83.19 (3.37)**	65.55 (0.74)	66.59 (0.44)	63.26 (11.47)	59.11 (9.04)	91.64 (0.73)	**98.45 (0.06)**	**99.14 (0.12)**	**86.07 (4.16)**	**99.55 (0.08)**	96.20 (0.11)	**62.55 (1.90)**

Table 14: PR-AUC comparison between Parametric MaskDiff-AD and matched NELBO scoring variants over five random seeds.

Method	Avg.	ad	aid362	apas	bank	census	chess	cmc	covertype	probe	r10	solar	u2r	vehicle	w7a
Parametric MaskDiff-AD	**32.39**	64.40 (3.96)	3.76 (0.94)	4.91 (1.17)	**23.90 (1.19)**	**9.06 (0.09)**	**0.69 (0.48)**	5.89 (3.37)	5.50 (0.83)	91.59 (0.68)	70.34 (3.75)	20.29 (6.55)	**46.37 (6.45)**	**93.04 (0.06)**	13.71 (2.47)
Fixed-grid NELBO	32.35	64.39 (4.00)	**3.82 (1.09)**	4.90 (1.20)	23.84 (1.06)	**9.06 (0.09)**	0.41 (0.30)	**6.76 (4.96)**	5.59 (0.85)	91.53 (0.66)	**70.46 (3.68)**	20.17 (7.16)	45.59 (6.37)	92.72 (0.08)	13.71 (2.44)
MC per-token NELBO	32.18	**64.51 (4.09)**	**3.82 (0.88)**	**4.95 (1.31)**	23.41 (1.13)	9.06 (0.05)	0.65 (0.57)	5.05 (2.99)	**5.60 (0.79)**	**91.61 (0.63)**	70.13 (3.60)	**20.40 (5.93)**	44.88 (5.96)	92.65 (0.10)	**13.74 (2.41)**