Memory Mechanism for Unsupervised Anomaly Detection

Jiahao Li[1,2] **Yiqiang Chen**[✉1,2] **Yunbing Xing**[1,2]

[1]Institute of Computing Technology, Chinese Academy of Sciences , Beijing, CN
[2]University of Chinese Academy of Sciences, Beijing, CN

Abstract

Unsupervised anomaly detection is a binary classification that detects anomalies in unseen samples given only unlabeled normal data. Reconstruction-based approaches are widely used, which perform reconstruction error minimization on training data to learn normal patterns and quantify the degree of anomalies by reconstruction errors on testing data. However, this approach tends to miss anomalies when the normal data has multi-pattern. Because the model generalizes unrestrictedly beyond normal patterns even to include anomaly patterns. In this paper, we proposed a memory mechanism that memorizes typical normal patterns through a capacity-controlled external differentiable matrix so that the generalization of the model to anomalies is limited by the retrieval of the matrix. We achieved state-of-the-art performance on several public benchmarks.

1 INTRODUCTION

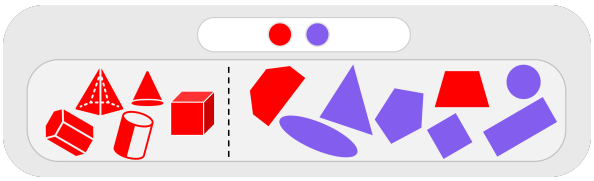
Overconfident models can lead to silent failures. Once a trained model is deployed into an open-world scenario, it will inevitably produce silent failures[González et al., 2022], meaning that the model is overconfident in subsuming unknown classes into known classes without making any declarations. The cost of silent failure is unacceptable in areas such as medical diagnosis, military decision-making, and financial risk control. Therefore, it is necessary to equip the model with the ability to truthfully report unknowns.

The unsupervised anomaly detection (UAD) task is kind of known-or-unknown judgment on unseen data given unlabeled known (normal) data, which requires the model to detect unknowns (anomalies) based on the generalization of the known (normal) data[Yang et al., 2021]. The reconstruction-based approach as shown in fig. 2 is the classical paradigm of UAD, which minimizes the reconstruction error on the normal data with the help of autoencoder (AE) framework for training, and then detects anomalies by reconstruction error[Bengio et al., 2006, Baldi, 2012, Ruff et al., 2021]. AD framework expect small reconstruction errors on normal samples and relatively large ones on anomaly samples. However, some studies have found and pointed out the failure case[Zong et al., 2018, Gong et al., 2019], i.e., the anomalies are also well generalized thus failing to produce significant reconstruction errors. To visualize the failure case of the overgeneralized anomalies, we show the illustration in fig. 3. The overgeneralized anomalies will lead to reconstruction errors that are difficult to distinguish from normal ones. The overgeneralization problem (OGP) has the following challenges.

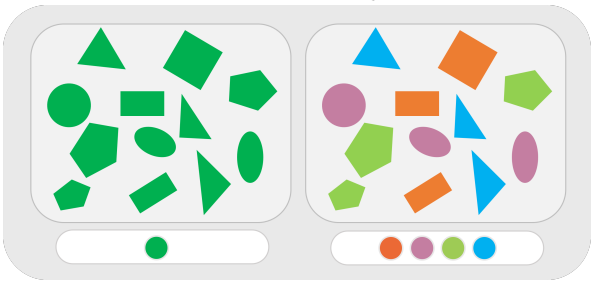
One challenge comes from the unlabeled training set, where the data may be non-single patterns. The lack of pattern labels leads to two dilemmas, as shown in fig. 1b. First, it is impossible to know what pattern an instance belongs to when given one. Second, it is impossible to know how many patterns the training data has when given one. In other words, neither the boundaries nor the number of patterns is available. Label-free guided AE networks need to generalize patterns in isolation. This is why unsupervised networks are unable to sensitively extract patterns in the data leading to overconfident models.

Another challenge comes from the test set, where the data may be semantic anomalies[Ahmed and Courville, 2019]. A semantic anomaly is an anomaly that differs from the normal pattern only at the semantic level. For example, in 2D graphical anomaly detection with known normal data, the anomalies in the test set can be roughly divided into two categories, as shown in fig. 1a. The two categories of anomalies on either side of the dotted line are 3D graphics and 2D graphics, respectively. For the anomaly detection model, detecting 3D anomalies is very simple because there is a big difference between 3D and 2D. However, detecting 2D anomalies requires further analysis of the number of edges, corners, and angles of the graph, which places a

Proceedings of the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023), PMLR 216:1219–1229.



(a) Test set challenge.



(b) Training set challenge.

Figure 1: Two major challenges of OGP. **Test Set Challenge:** Anomalies are represented in red and normal in purple. The anomalies shown on the left side of the dashed line are clearly different from the training set and can be detected easily. The right side of the dashed line shows the semantic anomalies. Hexagons and trapezoids represent semantic-level anomalies that are homologous to the original dataset, which is more insidious and difficult to detect. A large percentage of non-semantic anomalies can yield seemingly good performance. But when all the anomalies in the test set are semantic anomalies, the performance of the model is exposed realistically. **Training Set Challenge:** The unsupervised learning dilemma in the unlabeled multi-pattern training set is shown on the left. The training sets for the unlabeled and labeled scenarios are shown on the left and right, respectively.

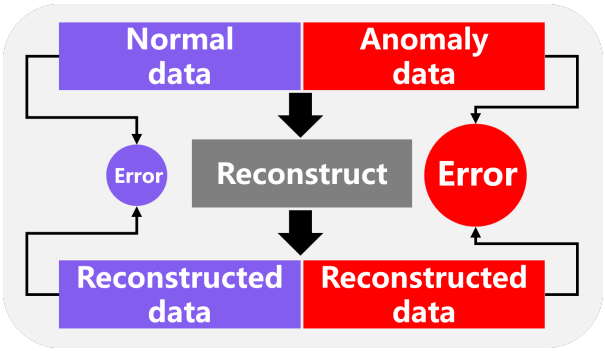


Figure 2: The reconstruction-based AD approaches.

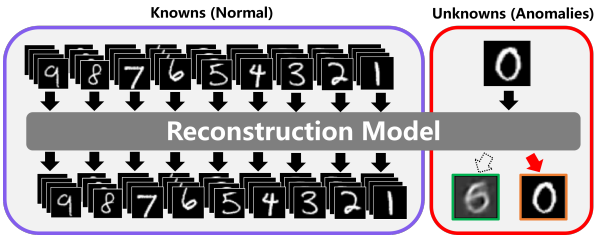


Figure 3: The visual illustration of overgeneralized anomalies. We selected the numbers 1 to 9 as the normal and 0 as the anomalies from MNIST. We use a fully connected AE as a reconstruction model to train on normal classes. The purple box in the figure shows the result of the model reconstructing the known classes. The lower right corner of the red box shows the reconstruction results of the anomalies. We expect the model to produce the reconstruction shown in the picture marked in the green box for normal classes, resulting in a sufficiently large reconstruction error. But in fact, the model outputs the picture marked by the orange box, which means that the model also generalizes well to the anomalies.

higher demand on pattern analysis at a finer granularity. In other words, semantic anomalies are more difficult to distinguish from normal ones.

Many methods have been proposed one after another to try to solve the OGP. MemAE[Gong et al., 2019] proposed a memory module that makes progress on a class of classification scenarios with a combination of prototype learning and sparse attention mechanisms. MNAD[Park et al., 2020] proposed a memory module that learns in a clustering-like manner without the aid of gradient updating. SSPCAB[Ristea et al., 2022] proposed a convolutional attention block to improve anomaly detection. It needs to be affirmed that the academic community has recognized that the reconstruction false-negative problem is caused by the model falling into the OGP, which means that the model simply generalizes a reconstruction constant mapping in a one-sided manner like memoryless learning.

However, no study has yet combined the two previously mentioned challenges(figs. 1a and 1b), i.e., detecting semantic anomalies under unlabeled multi-pattern normal data. Beyond that, all the existing methods inevitably introduce multiple hyperparameters both in the module and penalty term, which makes the models need to be well-tuned for deployment in real industrial applications in advance. And once the task is changed, the previous optimal combination of hyperparameters may need to be tuned again in order to achieve usable performance. These methods are not user-friendly in terms of comprehensibility and implementation in practical deployments. We propose a memory mechanism that can be performed simultaneously with model training and does not introduce any additional penalty term in the reconstruction loss. The memory mechanism can be well encapsulated by platforms such as PyTorch[Paszke et al., 2019] with only one line of code to equip existing models. The memory mechanism can effectively cope with the coexistence of training and test set challenges because the memory space learns to extract data patterns instead of just unilaterally generalizing the reconstruction mapping.

Overall, the core contributions of this paper are as follows:

- We proposed a capacity-controlled memory mechanism with a mapping-sharing strategy (section 3.2), which could be a viable solution to the OGP to cope with the coexistence of training and test set challenges.
- We proposed a memory-based autoencoder, called a Memorizer (section 3.4), which uses a multi-round memory mechanism for learning (section 3.3).
- We proposed a challenging experimental setup under the unlabeled non-single class normal data condition (section 3.1) conforming to real-world scenarios (fig. 1) different from the previous work with a non-single class of normal data, following the latest recommendations from academia.
- We reached state-of-the-art on several public benchmarks, proving the effectiveness of the memory mechanism.

2 RELATED WORK

Anomaly Detection. Anomaly detection is a complex problem because anomalies are unknown and rare[Pang et al., 2021]. Anomaly detection has been intensively studied under statistical techniques, such as Gaussian method[Barnett, 1976, Barnett and Lewis, 1984, Beckman and Cook, 1983, Ye and Chen, 2001], mixed parameter distributions method[Lauer, 2001, Eskin, 2000, Abraham and Chuang, 1989, Box and Tiao, 1968, Agarwal, 2005], histograms method[Eskin, 2000, Denning, 1987, Helman and Bhangoo, 1997], kernel functions method[Yeung and Chow, 2002, Bishop, 1994], and so on. However, these methods cannot effectively deal with high-dimensional data. With the development of deep learning techniques, deep anomaly detection models emerged[Chalapathy and Chawla, 2019]. Supervised methods are built on the basis that each normal class instance has a class label[Shilton et al., 2013, Jumutc and Suykens, 2014, Kim et al., 2015, Erfani et al., 2017]. However, such precisely labeled data for a mount of normal instances is often not available[Chalapathy and Chawla, 2019]. In contrast, unsupervised methods do not require data labeling but also face the following challenges[Chalapathy and Chawla, 2019, Gong et al., 2019, Zong et al., 2018]. First, learning the commonality of normal data in high-dimensional space. Second, how to choose the hyperparameters of the autoencoder to obtain optimal performance. Third, the autoencoder suffers from the OGP and fails to produce large reconstruction errors for anomalies.

Representation Learning. Several studies in recent years have been devoted to addressing the shortcomings of the unsupervised approach. Memory-based approaches are seen as promising solutions. MemAE[Gong et al., 2019] proposed memory modules that use the encoder output of the latent space to apply attention mechanisms to the memory prototypes in the module to obtain weights. The sparse weights are then used to weigh and sum the memory prototypes as decoder inputs. The weight sparsification loss is introduced as a penalty term in the loss function. MNAD[Park et al., 2020] draws on KMeans clustering[Lloyd, 1982] to update the memory prototype with a non-gradient style. The compact loss and separation loss are introduced into the loss function as penalty terms. SSPCAB[Ristea et al., 2022] proposed a masked convolution and attention block to improve anomaly detection. TrustMAE[Tan et al., 2021] proposed the concept of trusted regions based on MemAE to further prevent the autoencoder suffering from the OGP. Six additional penalty terms are introduced into the loss function. In summary, the improvement of memory shows a trend of more and more penalty terms and more complex structures. Excellent performance is constantly broken but the number of hyperparameters and module complexity is increasing. Is there a simple and elegant structure that can achieve good performance without introducing penalty terms and numerous hyperparameters? More relevant studies are waiting to be conducted.

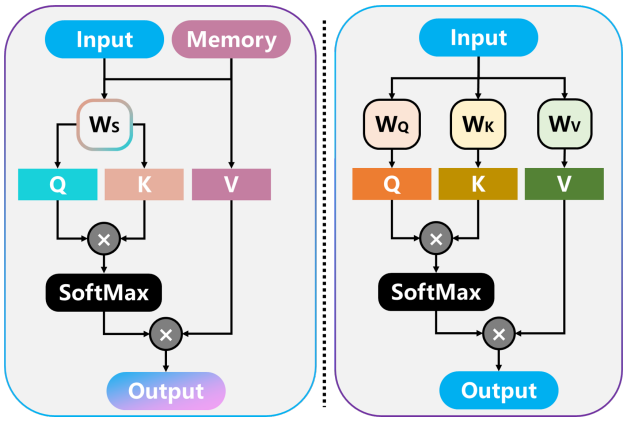


Figure 4: Memory versus Attention. On the left is our proposed memory mechanism and on the right is the self-attention mechanism.

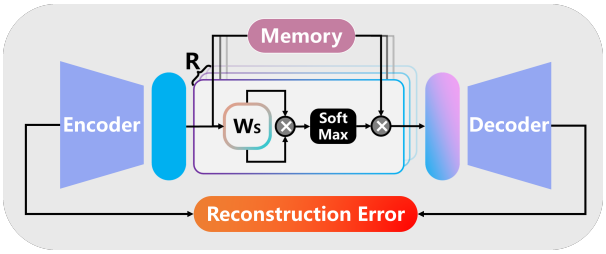


Figure 5: Overview of Memorizer.

Attention Mechanisms. The early success of the attention mechanism in the field of machine translation is unprecedented[Bahdanau et al., 2014], which is a technique that uses the query to compare keys to obtain weights to weigh

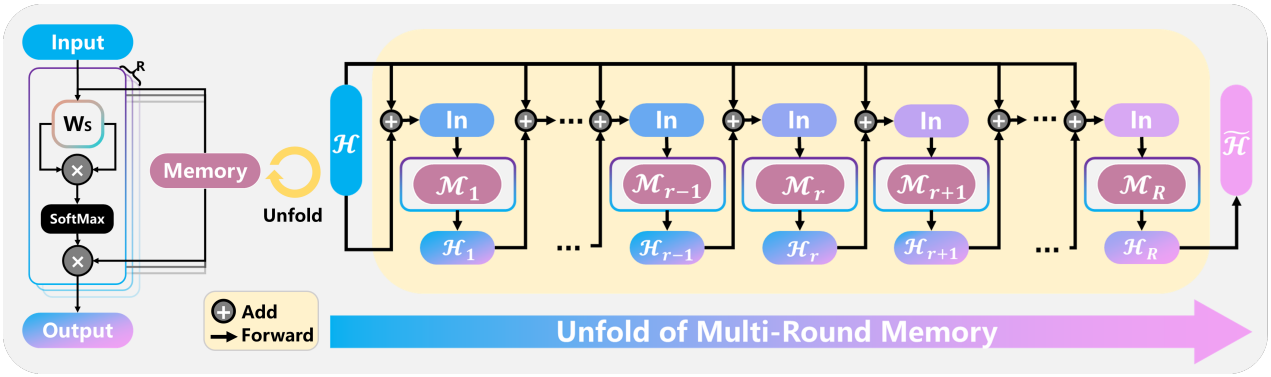


Figure 6: Multi-Round Memory.

the sum of values. After this, various variants of the attention mechanism emerged. General attention proposed trainable mapping matrices[Luong et al., 2015]. Hard attention proposed the concept of stochastic key[Xu et al., 2015]. The self-attention mechanism proposed to rely only on itself for attention operations[Yang et al., 2016]. Transformer proposed a multi-headed attention mechanism and obtained a breakthrough in the field of computer vision[Vaswani et al., 2017]. SENet proposed an attention mechanism for feature map channels[Hu et al., 2018]. More and more research is going deeper with the application of attention mechanisms and improvements in Transformers. The attentional mechanism and its variants show outstanding generalizability in experimental results. Can we do the opposite by using attention mechanisms to suppress overgeneralization to alleviate the OGP in UAD? More variants are to be studied.

3 METHODOLOGY

The main goal of the memory mechanism is to solve the problem of overgeneralization of unlabeled non-single-class data in UAD. fig. 4 illustrates the memory mechanism and how it compares to the self-attention mechanism. The attention mechanism mainly consists of mapping shared strategies and independent capacity-controlled memory, which are described in detail in section 3.2. fig. 6 demonstrates the multi-round memory structure based on the memory mechanism, details of which are expanded in section 3.3. The memory-based autoencoder called Memorizer is shown in fig. 5 and described in detail in section 3.4.

3.1 DEFINITION OF THE OVERGENERALIZATION PROBLEM (OGP)

Given a training set D, it is known that $D = \{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}$ contains only normal data, where (x, y) is the sample-label pair, n is the number of samples, and d is the dimension of sample $\forall x_i \in D, x_i \in \mathbb{R}^d$. Consider the label set $Y = \{y_i\}_{i=1}^{n}$ of dataset D, which is divided into ξ classes, i.e., $Y = \bigcup_{k=1}^{\xi} \{y_j | y_j = y_{c_k}, \forall j \in [1, n]\}$. Given a test set $T = \{(\widetilde{x}_1, \widetilde{y}_1), (\widetilde{x}_2, \widetilde{y}_2), \cdots, (\widetilde{x}_{n'}, \widetilde{y}_{n'})\}$ containing both normal and anomalous, where $(\widetilde{x}, \widetilde{y})$ is the sample-label pair, and n' is the number of samples $\forall \widetilde{x}_i \in T, \widetilde{x}_i \in \mathbb{R}^d$. The label set $\widetilde{Y} = \{\widetilde{y}_i\}_{i=1}^{n'}$ of T is divided into two classes as positive and negative respectively, i.e., $\widetilde{Y} = \bigcup_{c_k \in \{pos, neg\}} \{\widetilde{y}_j | \widetilde{y}_j = y_{c_k}, \forall j \in [1, n']\}$, where negative means normal and positive means anomaly. The OGP is defined as follows. Under the unlabeled non-single class normal data(UNSCND) condition $\{\xi > 1, Y = \emptyset\}$, the performance Q of the reconstruction-based model Ω on T decreases as the number of classes ξ or model capacity $\mathcal{O}(\Omega)$ increases, i.e., $Q_\Omega(T) \propto (\xi \mathcal{O}(\Omega))^{-1}$.

To truthfully and exclusively study the OGP without generating embellished scores in the experimental results that obscure the OGP, two key points needed to be stated. First, following the recommendation of paper[Ahmed and Courville, 2019] for the academic community, the anomaly should be semantic level, i.e., D and T come from the **SAME** dataset as shown on the right side of the dashed line in fig. 1. Second. it is important to note the distinction between traditional unlabeled single class normal data (USCND) condition $\{\xi = 1, Y = \emptyset\}$ and **UNSCND** condition $\{\xi > 1, Y = \emptyset\}$, as the smaller ξ tends to obscure the OGP in $Q_\Omega(T)$.

3.2 MEMORY MECHANISM

The input of the memory mechanism is denoted as $\mathcal{H} \in \mathbb{R}^{B \times (W*H*C)}$, where B, W, H, C are the batch size, width, height, and the number of channels, respectively. The main component of the memory mechanism consists of a trainable matrix $\mathcal{M} \in \mathbb{R}^{N \times F}$, where N is the memory capacity and $F = W*H*C$. $W_S \in \mathbb{R}^{F \times F}$ is a trainable linear mapping shared by $\mathcal{H}$ and $\mathcal{M}$. Denote the SoftMax[Bridle, 1989] function as σ in the direction of dimension N. The output $\widetilde{\mathcal{H}}$ of the memory mechanism is defined by eq. (1) as follows.

$$\widetilde{\mathcal{H}} = Memory\left(\mathcal{H}, \mathcal{M}\right) = \sigma\left(\mathcal{H}W_S\left(\mathcal{M}W_S\right)^T\right)\mathcal{M} \quad (1)$$

Notice that $\widetilde{\mathcal{H}} \in \mathbb{R}^{B\times F}$ in eq. (1) is in the same space as $\mathcal{H}$, and $\mathbb{R}^{B\times(W*H*C)}$ is the flattened form of $\mathbb{R}^{B\times H\times W\times C}$.

3.3 MULTI-ROUND MEMORY

In addition to the mapping sharing strategy W_S mentioned by section 3.2 to pull $\mathcal{H}$ and $\mathcal{M}$ into the same space for comparison, we found that multiple rounds of memory for the memory matrix $\mathcal{H}$ are also beneficial for overgeneralization suppression. Multi-round memory of learning helps memory to extract and consolidate the intrinsic patterns of the data in the form described by eq. (2).

$$\widetilde{\mathcal{H}}_r = Memory\left(\widetilde{\mathcal{H}}_{r-1} + \mathcal{H}, \mathcal{M}_r\right), \ \ (\widetilde{\mathcal{H}}_0 = \mathcal{H}) \quad (2)$$

$\widetilde{\mathcal{H}}_r$ and $\mathcal{M}_r(\forall r \in [1, R])$ represents the r-th round of $\mathcal{H}$ and $\mathcal{M}$ in the serial R-round memory learning. The output $\mathcal{H}_R$ of the final R-round memory is differentiable with respect to the first-round input $\mathcal{H}$ as described in eq. (3).

$$\begin{aligned}\frac{\partial\widetilde{\mathcal{H}}_R}{\partial\mathcal{H}} &= \frac{\partial\widetilde{\mathcal{H}}_R}{\partial\widetilde{\mathcal{H}}_0} \\ &= \prod_{i=1}^{R}\frac{\partial\widetilde{\mathcal{H}}_i}{\partial\widetilde{\mathcal{H}}_{i-1}} \\ &= \frac{\partial\widetilde{\mathcal{H}}_R}{\partial\widetilde{\mathcal{H}}_{R-1}}\frac{\partial\widetilde{\mathcal{H}}_{R-1}}{\partial\widetilde{\mathcal{H}}_{R-2}}\cdots\frac{\partial\widetilde{\mathcal{H}}_1}{\partial\widetilde{\mathcal{H}}_0}\end{aligned} \quad (3)$$

It can be noted that the final output after the R-th round of memory can be back-propagated to the encoder by the chain derivative law for the gradient, which ensures that the multi-round memorization process is differentiable.

3.4 MEMORIZER: MEMORY-BASED AUTO-ENCODER

Memorizers are composed on an autoencoder framework. Encoder $f_{\theta_E}: \mathbb{R}^d \to \mathbb{R}^F$ and decoder $g_{\theta_D}: \mathbb{R}^F \to \mathbb{R}^d$ are nonlinear learning functions, respectively. The structure of the Memorizer is described in eqs. (4) to (6).

$$\mathcal{H} = f_{\theta_E}\left(X\right) \quad (4)$$

$$\widetilde{\mathcal{H}} = \left(Memory\left(\widetilde{\mathcal{H}}_{r-1} + \mathcal{H}, \mathcal{M}_r\right)\right)_{r=1:R} \quad (5)$$

$$\widetilde{X} = g_{\theta_D}\left(\widetilde{\mathcal{H}}\right) \quad (6)$$

The eqs. (4) and (6) in which $X = \left\{x_{i_j}\right\}_{j=1}^{B} \in \mathbb{R}^{B\times d}$ denotes the input data of one batch and $\widetilde{X}$ denotes the reconstructed data. The loss function of the Memorizer is as follows.

$$L = \frac{1}{B}\sum_{j=1}^{B}\left\|x_{i_j} - \widetilde{x}_{i_j}\right\|_2^2 \quad (7)$$

x_{i_j} represents the data in D that is shuffled into the batch. Notice that the decoder input $\widetilde{H}$ of the memorizer comes from a linear weighted sum of $\widetilde{M}$. Therefore, the model generalizability is suppressed by and only by the memory capacity N and no longer depends on the model capacity $\mathcal{O}\left(\Omega\right)$. So the essence of the Memorizer is a controlled transformation of the OGP using the memory mechanism as described in eq. (8).

$$Q_\Omega\left(T\right) \propto \left(\xi\mathcal{O}\left(\Omega\right)\right)^{-1} \Rightarrow Q_{\mathcal{M}}\left(T\right) \propto \frac{\gamma}{\xi\mathcal{O}\left(\mathcal{M}\right)} \quad (8)$$

Notice that $\mathcal{O}\left(\mathcal{M}\right)$ is a correlation function on N. Without loss of generality, $\exists\delta, \gamma \in \mathbb{Z}^+$, $\lim\limits_{R\to\gamma} Q_{\mathcal{M}}\left(T\right) = \delta$ holds for fixed ξ and N.

4 EXPERIMENT

In this section, we conduct parallel comparison experiments for the OGP on multiple public benchmarks to verify the effectiveness of the Memorizer for overgeneralization inhibition with the principle of absolute fair comparison. Finally, we performed an ablation study and sensitivity analysis to ensure that the mechanisms and structures proposed in this paper are positive for solving the OGP.

4.1 EXPERIMENTAL SETUP

Datasets. The following three public benchmarks were used for the experiments in this paper, namely MNIST[Deng, 2012], Fashion[Xiao et al., 2017], and Kuzushiji[Clanuwat et al., 2018]. In order to make the dataset conform to the UAD setting and highlight OGP, they were all preprocessed under UNSCND settings (section 3.1), i.e., remove-one-class-out (ROCO) protocol as described below. For all classes in the dataset, we select one class to be removed from the training set and label that class as positive in the test set. For the remaining classes, we removed their label information from the training set and labeled them uniformly as negative in the test set. Note that the ROCO preprocessing is different from the USCND settings (section 3.1) of previous research [Gong et al., 2019, Park et al., 2020, Ristea et al., 2022], but rather the UNSCND condition mentioned in section 3.1, which is in line

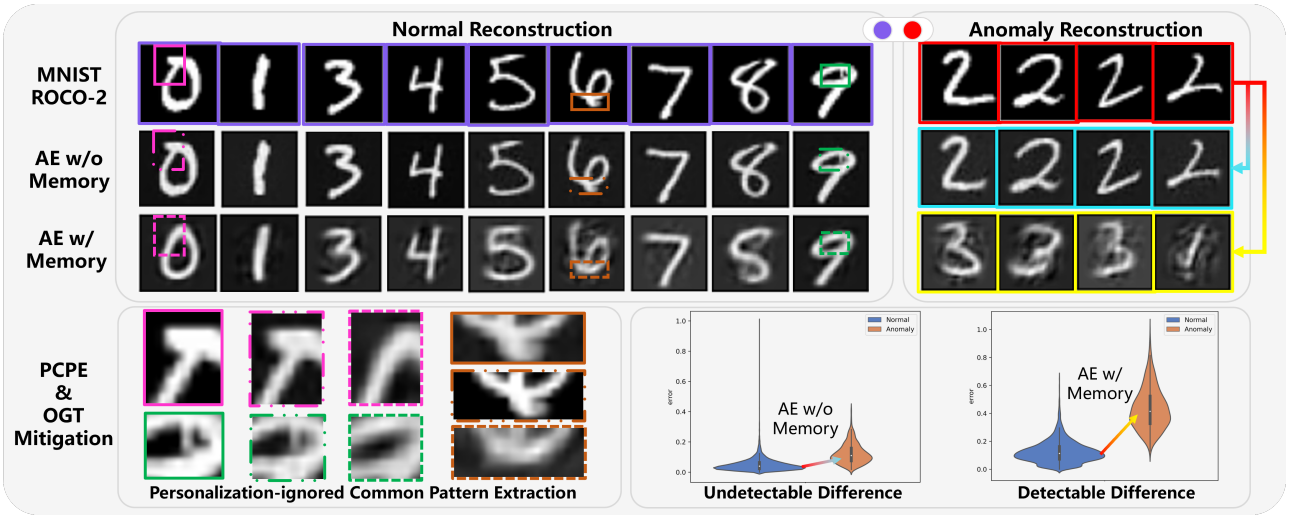


Figure 7: The memory mechanism alleviates the OGP in the face of the combined challenge of the training and test sets.

with the latest experimental setup recommendation from paper[Ahmed and Courville, 2019]. Also, this is consistent with the purpose of highlighting the OGP in UAD, i.e., the UNSCND condition of $\{\xi = 9, Y = \emptyset\}$.

Evaluation. AUROC is commonly used as an assessment criterion for anomaly detection, however, this is overly optimistic on unbalanced data[Movahedi et al., 2020]. We add results on the AUPRC and F1-Score evaluation criteria to examine the full range of model performance. F1-Score results use the best one after traversal threshold. Vanilla Autoencoder(AE)[Kramer, 1991], Sparse Autoencoder(SAE)[Ng et al., 2011], Denoising Autoencoder(DAE)[Vincent et al., 2008], Variational Autoencoder(VAE)[Kingma and Welling, 2013], MemAE[Gong et al., 2019], MNAD[Park et al., 2020], and SSPCAB[Ristea et al., 2022] were used as comparative baselines.

Implementation. To facilitate the description, we make the following notation convention. FC(a,b) denotes the Linear-BN-LeakyReLU block, where a and b are the number of input and output channels. We use the same autoencoder backbone in all experiments, i.e., FC1(784,512), FC2(512,256), FC3(256,128), FC4(128,256), FC5(256,512), FC6(512,784). Note that since MNAD requires a residual structure at the bottleneck, the first layer of the MNAD decoder is twice as large as the other models, namely: FC4(128+128,256). The memory capacity of the three datasets MNIST, Fashion, and Kuzushiji is 8, 10, and 10 in order, considering that the latter two have more complex data patterns. The experiments used a batch size of 256, an optimizer Adam with a learning rate of 1e-3, multi-round memory with rounds of 8, and an early stop mechanism with the patience of 10. The validation set split ratio was 0.1 and the split random seeds were fixed with 2022. The experimental results were averaged over three runs. All comparison experiments follow the principle of absolute fair comparison(PAFC), i.e., all experiments strictly use the same training set, test set, validation set, and backbone.

4.2 RESULTS

We designed a total of 30 experiments using 3 public benchmarks and conducted 8 parallel comparisons at 3 different evaluation metrics based on PAFC. Our average results achieve the leading performance as shown in tables 1 to 3. The best performance is marked in bold.

4.3 ANALYSIS

The Distinguishability of Memory We normalized the reconstruction errors of the model on normal and anomaly data separately for better comparison and depicted their KDE distribution as shown in figs. 7 and 8. It can be seen that the vanilla autoencoder (AE) suffers from OGP when faced with the dual challenge of the test set and training set, while the AE equipped with memory can better distinguish between normal and anomaly. The ultimate goal of anomaly detection is to score the abnormalities of a sample, which comes directly from reconstruction errors. Therefore a distinguishable model has a sharper ability to score abnormalities.

The User-friendliness of Memory Notice that VAE also has good performance, but two more points need to be highlighted. First, the results of VAE come from a fine grid search and careful tuning of the parameters, while the memorizer just uses the default parameters without any

Table 1: The AUROC results under ROCO protocol.

MNIST	AE	SAE	DAE	VAE	MemAE	MNAD	SSPCAB	Ours
0	79.16±0.30	82.57±0.44	82.86±1.06	94.68±1.73	86.16±1.2	79.88±0.33	80.95±0.12	**97.79±0.64**
1	12.45±0.94	19.21±3.36	14.05±0.99	38.99±0.40	17.16±2.93	16.90±1.43	11.39±0.58	**41.61±0.03**
2	86.20±0.22	89.26±0.15	89.23±1.76	97.14±0.08	93.80±1.35	86.63±0.64	86.80±0.02	**97.95±0.27**
3	64.68±0.10	67.91±3.88	68.62±2.51	95.11±0.07	82.05±3.12	66.62±0.41	66.64±0.24	**94.97±0.61**
4	59.13±0.45	66.88±0.51	62.34±3.68	92.80±0.12	77.04±3.14	65.29±4.48	59.78±0.48	**89.84±0.81**
5	71.34±0.15	75.78±0.73	73.77±3.06	**96.45±0.09**	81.16±2.69	72.13±0.62	71.56±0.76	96.21±0.18
6	84.92±0.30	82.87±0.36	86.82±1.96	93.46±0.13	89.64±0.52	85.61±1.49	84.80±0.36	**94.97±0.44**
7	61.91±0.85	58.75±1.91	62.93±1.97	74.92±1.18	61.20±3.09	67.45±2.13	59.79±0.43	**84.14±1.05**
8	67.84±0.16	76.32±0.81	73.22±4.39	96.02±0.38	91.06±0.54	75.90±1.58	69.22±0.09	**96.16±0.73**
9	44.38±1.70	47.90±6.11	43.79±1.02	71.61±0.04	53.54±3.41	51.29±2.24	44.69±1.04	**78.99±0.70**
AVG	63.20±0.52	66.75±1.83	65.76±2.24	85.12±0.42	73.33±2.20	66.77±1.53	63.56±0.41	**87.26±0.55**
Fashion	AE	SAE	DAE	VAE	MemAE	MNAD	SSPCAB	Ours
T-shirt	57.03±0.24	44.28±2.63	58.08±1.00	58.34±0.22	55.66±0.92	57.15±1.99	57.63±0.12	**60.33±0.08**
Trouser	71.03±1.25	88.37±1.25	73.80±2.22	84.94±0.18	**88.78±0.48**	70.06±2.11	63.66±1.35	84.28±0.37
Pullover	43.64±0.25	33.31±0.21	47.41±1.24	58.58±0.19	52.75±0.41	45.19±1.13	43.76±0.16	**58.71±0.10**
Dress	61.30±0.97	65.09±1.48	63.79±2.24	**70.84±0.42**	67.52±1.16	63.47±1.87	64.48±0.58	70.61±0.27
Coat	48.93±0.16	45.47±4.27	51.31±0.25	54.98±0.28	52.32±1.50	48.74±0.21	49.95±0.32	**56.35±0.11**
Sandal	92.28±0.46	**92.63±0.30**	90.87±1.14	87.38±0.29	89.25±0.34	91.85±0.92	91.70±0.23	86.17±0.19
Shirt	51.15±0.52	34.04±1.53	50.23±0.54	**54.47±0.03**	52.63±0.96	51.20±0.53	51.59±0.20	54.23±0.35
Sneaker	64.85±0.40	**70.58±3.16**	61.49±2.84	65.22±0.52	60.66±1.26	61.75±2.77	61.62±0.49	64.01±0.52
Bag	95.56±0.17	88.74±0.29	96.18±0.51	94.07±0.26	94.84±0.36	96.13±0.18	**96.39±0.17**	95.15±0.29
Ankle boot	83.41±0.64	83.36±0.47	84.71±0.78	77.28±0.01	82.22±0.65	83.78±0.61	**86.73±0.58**	81.32±0.57
AVG	66.92±0.51	64.59±1.56	67.79±1.28	70.61±0.24	69.66±0.8	66.93±1.23	66.75±0.42	**71.12±0.29**
Kuzushiji	AE	SAE	DAE	VAE	MemAE	MNAD	SSPCAB	Ours
U+304A	71.50±0.03	71.11±2.57	71.36±0.86	81.42±0.33	76.92±0.18	71.96±0.05	71.56±0.35	**85.14±0.43**
U+304D	50.33±0.28	51.36±0.49	49.25±0.40	68.00±0.15	60.88±0.76	50.66±0.24	50.00±0.07	**71.04±0.03**
U+3059	43.32±0.07	46.93±0.31	42.95±0.42	59.05±0.50	52.05±0.54	44.41±0.16	43.18±0.09	**63.14±0.2**
U+3064	68.78±0.17	72.65±0.54	69.96±0.41	78.83±0.01	73.77±1.46	69.62±0.24	69.36±0.04	**82.37±0.55**
U+306A	57.31±0.08	61.51±0.76	60.10±1.07	83.86±0.38	77.46±1.05	56.61±1.29	58.08±0.07	**87.38±0.28**
U+306F	19.35±0.17	22.19±1.15	20.51±0.93	52.66±0.45	37.40±0.05	19.90±0.27	20.08±0.26	**61.14±0.74**
U+307E	39.85±0.07	45.03±1.61	40.75±0.36	65.23±0.40	55.94±0.17	39.16±0.05	40.07±0.25	**68.24±0.49**
U+3084	80.46±0.10	80.79±0.58	81.46±0.57	90.33±0.42	88.66±0.27	80.80±0.86	80.79±0.05	**92.11±0.07**
U+308C	62.19±0.27	59.89±0.92	60.38±0.46	68.26±0.03	64.37±0.34	63.99±0.26	61.11±0.60	**73.01±0.91**
U+3092	65.95±0.05	63.24±1.00	66.65±0.70	78.07±0.24	73.44±0.42	66.45±0.11	66.43±0.11	**81.41±0.55**
AVG	55.90±0.13	57.47±0.99	56.34±0.62	72.57±0.29	66.09±0.52	56.36±0.35	56.07±0.19	**76.50±0.42**

deliberate tuning. Second, as described in the paragraph above the contributions section(section 1), VAE models face posterior collapse problems if it is not well-tuned, as shown in fig. 9, whereas Memory has no such concerns.

The Tightness of Memory As described in section 1, the model can both generalize well to the normal instance and fail to generalize anomalies under ideal assumptions, which require the model to learn tight bounds on the data patterns. As shown in fig. 7, the model was never exposed to the number 2 during training in the MNIST ROCO-2 experiment, but AE still reconstructed it well. This phenomenon indicates that AE does not learn the tight boundaries of patterns, but only one-sidedly learns the generalized constant mapping. In contrast, the AE that used the memory mechanism memorized the pattern in the training phase and followed the known patterns for the reconstruction of the unknown category in the testing phase.

The Diversity of Memory In order to generalize data with limited memory capacity and learn patterns with tight boundaries, AE equipped with memory extract invariant features under the same class of data, i.e., Personalization-ignored Common Pattern Extraction (PCPE) process as shown in fig. 7. PCPE helps to learn the common features of the same pattern and ignore the semantic redundant features such as: starting position, stopping position, etc., so as to better establish the tight boundaries of the pattern. However, PCPE does not mean that memory let the model lose the intra- and inter-pattern diversity of the reconstruction as shown in fig. 10. We generated the new representation by generating random numbers in the range $[0, 1]$ to simulate the memory combination coefficients, and after going through the decoder we can see that the diversity between patterns is guaranteed.

Table 2: The AUPRC results under ROCO protocol.

MNIST	AE	SAE	DAE	VAE	MemAE	MNAD	SSPCAB	Ours
0	25.83±0.69	35.59±0.66	31.61±1.87	62.26±0.53	39.98±3.23	26.50±0.61	28.75±0.34	**80.24±1.52**
1	6.27±0.04	6.67±0.24	6.37±0.06	8.47±0.05	6.57±0.16	6.51±0.08	6.23±0.04	**8.81±0.01**
2	44.93±0.52	56.38±1.92	53.31±5.13	83.11±0.36	67.98±6.65	46.16±1.21	46.10±1.12	**86.06±0.68**
3	16.84±0.22	17.96±3.65	19.02±1.82	**64.20±0.85**	31.06±4.58	17.31±0.27	17.62±0.18	61.25±4.30
4	13.24±0.25	16.54±0.86	14.20±1.41	**60.84±0.04**	25.84±3.25	16.44±2.61	14.13±0.42	48.88±0.46
5	16.26±0.13	19.92±0.74	18.16±2.11	**73.92±0.33**	25.80±3.49	17.08±0.40	16.60±0.62	69.30±0.49
6	41.85±0.83	35.49±0.74	48.60±4.87	66.06±0.70	54.59±1.00	42.96±2.49	43.48±0.74	**70.68±0.54**
7	19.13±0.69	13.91±0.75	21.06±1.76	25.22±1.84	15.15±1.25	23.09±1.66	19.64±0.31	**36.51±0.27**
8	14.98±0.06	23.62±0.93	18.84±2.88	**72.64±1.38**	49.03±2.02	20.97±1.42	16.06±0.17	66.82±3.82
9	8.82±0.31	9.86±1.61	8.71±0.18	**17.81±0.06**	10.31±0.76	9.98±0.49	8.69±0.10	26.78±0.35
AVG	20.81±0.37	23.59±1.11	23.99±2.21	53.45±1.61	32.63±2.64	22.70±1.12	21.73±0.40	**55.53±1.24**
Fashion	AE	SAE	DAE	VAE	MemAE	MNAD	SSPCAB	Ours
T-shirt	11.57±0.03	9.68±0.50	11.99±0.19	12.99±0.05	11.62±0.19	11.67±0.72	11.74±0.12	**13.7±0.12**
Trouser	15.37±0.50	**39.17±3.44**	16.87±1.24	31.17±0.08	37.73±1.33	14.76±1.15	12.25±0.43	29.00±0.80
Pullover	8.91±0.03	7.41±0.08	9.39±0.17	11.91±0.11	10.25±0.09	9.03±0.20	8.76±0.03	**11.92±0.08**
Dress	12.32±0.28	14.15±0.56	12.87±0.59	**16.65±0.33**	14.27±0.44	12.90±0.73	13.18±0.17	15.91±0.17
Coat	8.94±0.03	8.64±0.74	9.34±0.05	10.37±0.04	9.64±0.26	8.87±0.05	9.09±0.07	**10.60±0.03**
Sandal	49.03±1.46	**52.72±1.07**	48.16±1.47	43.19±0.81	44.75±0.55	49.71±2.24	51.87±0.79	39.48±0.12
Shirt	11.34±0.15	7.64±0.34	10.93±0.22	10.90±0.05	11.08±0.22	11.12±0.13	**11.37±0.08**	10.94±0.06
Sneaker	12.33±0.14	**14.72±1.36**	11.38±0.82	12.88±0.23	11.20±0.32	11.45±0.73	11.33±0.14	12.27±0.12
Bag	66.06±1.29	43.59±0.50	69.71±2.62	56.53±1.37	58.42±2.10	69.47±1.26	**75.12±0.76**	64.72±2.23
Ankle boot	27.69±0.54	31.08±0.92	31.25±1.93	22.42±0.13	27.30±0.81	28.93±1.36	**33.29±0.91**	30.52±1.37
AVG	22.36±0.45	22.88±0.95	23.19±0.93	22.90±0.32	23.63±0.63	22.79±0.86	23.80±0.35	**23.91±0.51**
Kuzushiji	AE	SAE	DAE	VAE	MemAE	MNAD	SSPCAB	Ours
U+304A	15.92±0.10	16.73±1.44	15.86±0.52	31.21±0.75	22.07±0.44	16.08±0.39	16.27±0.95	**37.93±1.23**
U+304D	9.45±0.04	9.82±0.08	9.33±0.08	17.65±0.26	13.65±0.54	9.57±0.09	9.40±0.03	**20.17±0.01**
U+3059	9.09±0.03	10.40±0.55	9.13±0.19	13.01±0.04	11.27±0.11	9.66±0.03	9.11±0.16	**15.04±0.15**
U+3064	20.23±0.17	24.64±0.48	20.40±0.53	29.75±0.25	25.52±1.58	21.36±0.64	20.39±0.39	**32.09±0.45**
U+306A	10.92±0.04	12.25±0.30	11.73±0.36	33.67±1.25	22.52±1.30	10.79±0.30	11.13±0.04	**42.99±0.55**
U+306F	5.81±0.01	6.03±0.08	5.89±0.07	9.66±0.06	7.30±0.02	5.83±0.01	5.85±0.02	**12.36±0.14**
U+307E	9.00±0.11	10.75±0.60	9.46±0.13	22.52±0.12	16.27±0.36	8.96±0.07	9.20±0.08	**26.15±0.62**
U+3084	25.26±0.27	24.35±0.40	26.39±0.64	53.26±2.44	46.60±0.25	25.56±1.31	25.55±0.08	**58.98±1.12**
U+308C	13.90±0.08	11.92±0.48	12.56±0.08	15.27±0.03	13.39±0.06	13.56±0.26	13.05±0.39	**18.16±0.70**
U+3092	15.01±0.12	13.97±0.48	15.45±0.46	27.36±0.49	19.81±0.39	15.51±0.40	15.48±0.08	**31.10±1.28**
AVG	13.46±0.10	14.09±0.49	13.62±0.31	25.34±0.57	19.84±0.50	13.69±0.35	13.54±0.22	**29.50±0.62**

Table 3: The best F1-Score under ROCO protocol.

MNIST	AE	SAE	DAE	VAE	MemAE	MNAD	SSPCAB	Ours
0	36.52±0.52	41.35±0.62	41.15±1.76	63.97±6.58	44.85±1.38	36.63±0.59	38.73±0.42	**80.16±0.39**
1	20.39±0.00	20.41±0.03	20.41±0.02	20.92±0.04	20.42±0.05	20.39±0.00	20.39±0	**22.78±0.06**
2	48.47±0.78	56.51±1.07	54.72±3.89	75.93±0.45	65.39±4.26	49.24±1.17	49.85±0.48	**78.40±0.94**
3	22.98±0.04	25.70±2.78	24.99±1.15	65.36±0.28	38.26±3.51	23.91±0.14	23.95±0.18	**65.52±0.84**
4	20.40±0.11	24.18±0.15	21.77±1.25	**59.84±0.03**	33.11±2.57	23.85±3.06	20.33±0.14	52.42±0.70
5	25.83±0.26	30.16±1.11	28.12±2.68	**70.32±0.29**	35.61±3.29	26.72±0.46	25.79±0.57	67.24±0.75
6	47.46±0.77	39.98±0.80	51.55±3.40	62.80±0.27	56.01±1.21	47.64±2.18	48.38±0.9	**66.39±0.48**
7	22.78±0.75	21.00±0.86	24.01±1.23	31.85±1.18	21.73±1.32	27.69±1.39	22.3±0.51	**42.97±0.32**
8	24.98±0.15	30.74±0.44	28.37±2.63	68.02±0.52	51.88±1.52	29.96±1.36	25.42±0.04	**70.27±0.94**
9	19.77±0.05	19.60±0.94	20.07±0.13	27.65±0.08	21.38±1.02	20.68±0.3	19.51±0.08	**36.26±0.31**
AVG	28.96±0.34	30.96±0.88	31.52±1.81	54.67±0.97	38.86±2.01	30.67±1.07	29.46±0.33	**58.24±0.57**
Fashion	AE	SAE	DAE	VAE	MemAE	MNAD	SSPCAB	Ours
T-shirt	20.80±0.12	18.63±0.33	21.21±0.47	20.77±0.04	20.21±0.37	20.95±0.7	20.91±0.06	**21.58±0.01**
Trouser	28.67±0.96	**49.32±2.22**	30.74±1.75	42.95±0.27	50.60±0.92	28.45±0.95	25.29±0.67	42.33±0.94
Pullover	18.81±0.14	18.18±0.00	19.30±0.19	21.36±0.04	20.24±0.09	18.95±0.12	18.87±0.09	**21.41±0.02**
Dress	23.18±0.39	24.71±1.20	24.31±1.04	**27.72±0.34**	25.79±0.63	24.10±0.72	24.65±0.41	27.40±0.34
Coat	20.33±0.01	18.57±0.08	20.79±0.09	20.91±0.04	20.81±0.47	20.29±0.07	20.76±0.05	**21.54±0.09**
Sandal	55.15±1.43	**56.79±0.65**	51.09±3.00	45.45±0.81	47.93±0.44	53.68±2.73	53.59±0.81	42.98±0.51
Shirt	19.05±0.08	18.21±0.00	18.87±0.08	**19.72±0.03**	19.45±0.18	19.18±0.06	19.20±0.06	19.65±0.02
Sneaker	26.77±0.19	**29.03±1.76**	25.52±1.11	25.16±0.02	24.31±0.65	25.46±1.18	25.85±0.12	25.04±0.26
Bag	70.06±0.78	52.57±0.45	71.56±1.23	63.09±0.68	65.35±1.10	71.45±0.17	**74.18±0.42**	66.85±1.46
Ankle boot	38.82±0.83	39.53±0.32	40.62±0.96	32.62±0.27	36.98±0.75	39.36±0.65	**43.88±0.83**	37.85±0.92
AVG	32.16±0.49	32.55±0.70	32.4±0.99	31.97±0.25	**33.17±0.56**	32.19±0.73	32.72±0.35	32.66±0.46
Kuzushiji	AE	SAE	DAE	VAE	MemAE	MNAD	SSPCAB	Ours
U+304A	28.78±0.06	28.17±1.92	28.39±0.55	37.54±0.39	37.54±0.39	29.01±0.23	28.91±0.07	**43.97±0.87**
U+304D	19.26±0.07	19.41±0.08	19.03±0.13	25.24±0.17	25.24±0.17	19.25±0.06	19.18±0.02	**27.17±0.17**
U+3059	18.40±0.03	18.88±0.06	18.44±0.06	21.09±0.24	21.09±0.24	18.35±0.03	18.37±0.03	**21.99±0.02**
U+3064	26.73±0.12	29.35±0.70	27.07±0.09	36.21±0.11	36.21±0.11	27.63±0.07	27.27±0.05	**39.50±0.86**
U+306A	21.44±0.07	22.72±0.28	22.47±0.43	42.02±0.96	42.02±0.96	21.08±0.76	21.65±0.03	**49.20±0.12**
U+306F	18.18±0.00	18.19±0.00	18.19±0.00	20.10±0.11	20.10±0.11	18.18±0.00	18.18±0.00	**22.45±0.40**
U+307E	18.18±0.00	18.19±0.00	18.22±0.02	24.78±0.23	24.78±0.23	18.20±0.01	18.21±0.02	**27.61±0.15**
U+3084	36.14±0.01	36.89±1.14	36.68±0.67	54.85±1.17	54.85±1.17	36.73±0.69	36.46±0.02	**57.92±0.19**
U+308C	22.60±0.01	21.85±0.27	21.84±0.23	26.44±0.10	26.44±0.10	23.60±0.23	22.10±0.34	**30.29±1.05**
U+3092	24.22±0.24	22.87±0.50	24.26±0.30	35.46±0.07	35.46±0.07	24.26±0.01	24.04±0.13	**39.02±0.63**
AVG	23.39±0.06	23.65±0.5	23.46±0.25	32.37±0.35	32.37±0.35	23.63±0.21	23.44±0.07	**35.91±0.45**

4.4 ABLATION STUDY

To investigate the role of each component of the memory mechanism, we conducted qualitative ablation experiments to explore the effects of the add operation, the softmax operation, the multi-round memory, and the sharing mapping on the model performance, respectively. Ablation experiments were carried out in a randomly selected Fashion under ROCO-6. The ablation results in table 4 illustrate that all four components mentioned above play an effective role in the memory mechanism to varying degrees.

4.5 SENSITIVITY ANALYSIS

The memory mechanism involves two hyperparameters, the number of memory rounds R and the memory capacity N.

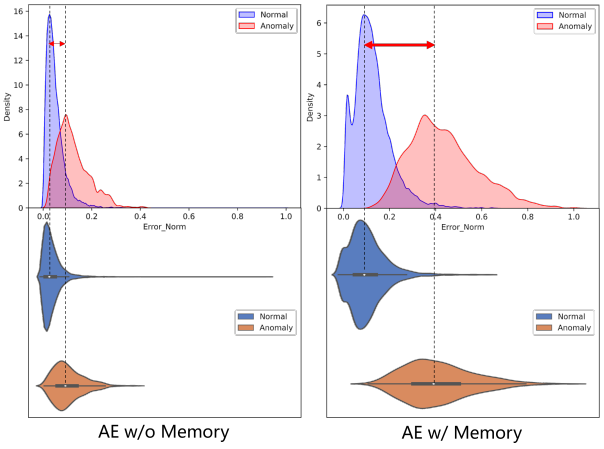


Figure 8: The distinguishability of memory.

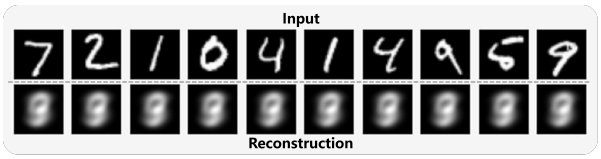


Figure 9: Posterior collapse of VAE.

We simply fixed $R = 8$ in all experiments and $N = 10$ in all experiments except for the MNIST capacity $N = 8$. The original intention of this setting was to expand the memory capacity because we thought that Fashion and Kuzushiji have more complex data patterns compared to MNIST. To understand in detail the effect of different parameters on model performance, we randomly selected MNIST ROCO-4 experiments for sensitivity analysis as shown in section 4.5. We did eight sets of experiments in the range of $[2, 16]$ at intervals of 2 for memory capacity N and memory round number R. Their AUROC scores (fig. 11) are represented by AUC(N) [Blue] and AUC(R) [Orange], respectively. It is easy to find that the memory capacity decreases the model performance when it is too small ($N = 2$), and the model performance changes relatedly as the capacity increases and reaches the optimum at a particular capacity ($N = 14$). In contrast, changes in the number of rounds R have less impact on AUROC, and the model performance remains stable.

5 CONCLUSIONS

We proposed a memory mechanism for UAD to address the dual challenges of label-free multi-pattern and semantic-level anomalies, which can be plug-and-play as a module for existing models without adding penalty terms. The Memorizer model equipped with multi-round memory can effectively alleviate the OGP in UAD and allow the models to report the unknowns truthfully.

Table 4: The ablation results.

Add	Softmax	Round	Sharing	AUC	AP	F1
	✓	✓	✓	52.61	9.80	21.07
✓		✓	✓	89.67	46.87	49.25
✓	✓		✓	88.98	46.07	48.29
✓	✓	✓		88.82	45.55	47.43
✓	✓	✓	✓	89.88	48.04	49.58

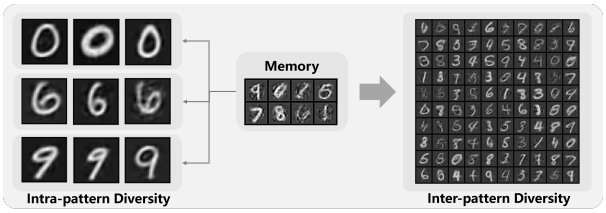


Figure 10: The diversity of memory.

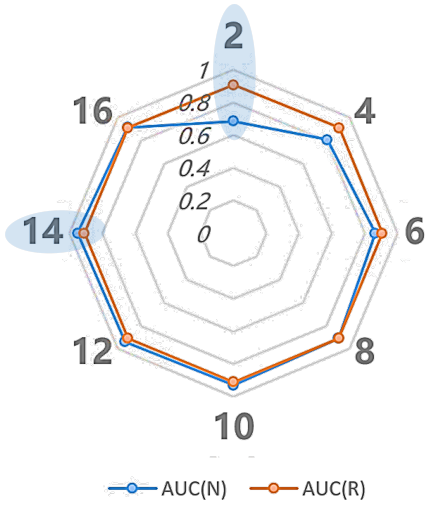


Figure 11: The sensitivity of memory.

6 FUTURE WORKS

The OGP proposed in this paper has significant implications for the study of real industrial production environment deployment in the future. Existing UAD methods need to train multiple models separately for multiple patterns in practical applications, e.g., cup anomaly detection model for cups, nail anomaly detection model for nails, and box anomaly detection model for boxes, which is a model flooding dilemma. The formulation of the OGP clarifies the model flooding dilemma and opens up a new research direction by proposing the UNSCND condition. In future work, the theorization and application of memory mechanisms are worthy of continued in-depth research. Further proof and derivation of the existence and approximation principles of tight bounds for unlabeled multi-pattern data are needed on the theoretical side. In terms of applications, the combination of memory mechanisms with continuous learning, domain generalization, and generative networks can be explored, which are all anticipated works.

Acknowledgements

This work is supported by the National Key Research and Development Program of China No.2018YFC2002603.

References

Bovas Abraham and Alice Chuang. Outlier detection and time series modeling. *Technometrics*, 31(2):241–248, 1989.

Deepak Agarwal. An empirical bayes approach to detect anomalies in dynamic multidimensional arrays. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*, pages 8–pp. IEEE, 2005.

Faruk Ahmed and Aaron C. Courville. Detecting semantic anomalies. *CoRR*, abs/1908.04388, 2019. URL `http://arxiv.org/abs/1908.04388`.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Pierre Baldi. Autoencoders, unsupervised learning, and deep architectures. In *Proceedings of ICML workshop on unsupervised and transfer learning*, pages 37–49. JMLR Workshop and Conference Proceedings, 2012.

Vic Barnett. The ordering of multivariate data. *Journal of the Royal Statistical Society: Series A (General)*, 139(3): 318–344, 1976.

Vic Barnett and Toby Lewis. Outliers in statistical data. *Wiley Series in Probability and Mathematical Statistics. Applied Probability and Statistics*, 1984.

Richard J Beckman and R Dennis Cook. Outlier..... s. *Technometrics*, 25(2):119–149, 1983.

Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, 19, 2006.

Christopher M Bishop. Novelty detection and neural network validation. *IEE Proceedings-Vision, Image and Signal processing*, 141(4):217–222, 1994.

George EP Box and George C Tiao. A bayesian approach to some outlier problems. *Biometrika*, 55(1):119–129, 1968.

John Bridle. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. *Advances in neural information processing systems*, 2, 1989.

Raghavendra Chalapathy and Sanjay Chawla. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*, 2019.

Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical japanese literature. *CoRR*, abs/1812.01718, 2018. URL `http://arxiv.org/abs/1812.01718`.

Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

D.E. Denning. An intrusion-detection model. *IEEE Transactions on Software Engineering*, SE-13(2):222–232, 1987. doi: 10.1109/TSE.1987.232894.

Sarah M Erfani, Mahsa Baktashmotlagh, Masud Moshtaghi, Vinh Nguyen, Christopher Leckie, James Bailey, and Kotagiri Ramamohanarao. From shared subspaces to shared landmarks: A robust multi-source classification approach. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Eleazar Eskin. Anomaly detection over noisy data using learned probability distributions. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, page 255–262, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. ISBN 1558607072.

Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1705–1714, 2019.

Camila González, Karol Gotkowski, Moritz Fuchs, Andreas Bucher, Armin Dadras, Ricarda Fischbach, Isabel Jasmin Kaltenborn, and Anirban Mukhopadhyay. Distance-based detection of out-of-distribution silent failures for covid-19 lung lesion segmentation. *Medical image analysis*, 82: 102596, 2022.

Paul Helman and Jessie Bhangoo. A statistically based system for prioritizing information exploration under uncertainty. *Ieee transactions on systems, man, and cybernetics-part a: Systems and humans*, 27(4):449–466, 1997.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

Vilen Jumutc and Johan AK Suykens. Multi-class supervised novelty detection. *IEEE transactions on pattern analysis and machine intelligence*, 36(12):2510–2523, 2014.

Sangwook Kim, Yonghwa Choi, and Minho Lee. Deep learning with support vector data description. *Neurocomputing*, 165:111–117, 2015.

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

Mark A Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal*, 37(2):233–243, 1991.

Martin Lauer. A mixture approach to novelty detection using training data with outliers. In *European Conference on Machine Learning*, pages 300–311. Springer, 2001.

Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

Faezeh Movahedi, Rema Padman, and James F Antaki. Limitations of roc on imbalanced data: Evaluation of lvad mortality risk scores. *arXiv preprint arXiv:2010.16253*, 2020.

Andrew Ng et al. Sparse autoencoder. *CS294A Lecture notes*, 72(2011):1–19, 2011.

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.

Hyunjong Park, Jongyoun Noh, and Bumsub Ham. Learning memory-guided normality for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14372–14381, 2020.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Nicolae-Cătălin Ristea, Neelu Madan, Radu Tudor Ionescu, Kamal Nasrollahi, Fahad Shahbaz Khan, Thomas B Moeslund, and Mubarak Shah. Self-supervised predictive convolutional attentive block for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13576–13586, 2022.

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.

Alistair Shilton, Sutharshan Rajasegarar, and Marimuthu Palaniswami. Combined multiclass classification and anomaly detection for large-scale wireless sensor networks. In *2013 IEEE eighth international conference on intelligent sensors, sensor networks and information processing*, pages 491–496. IEEE, 2013.

Daniel Stanley Tan, Yi-Chun Chen, Trista Pei-Chun Chen, and Wei-Chao Chen. Trustmae: A noise-resilient defect classification framework using memory-augmented auto-encoders with trust regions. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 276–285, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103, 2008.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015.

Jie Yang, Ruijie Xu, Zhiquan Qi, and Yong Shi. Visual anomaly detection for images: A survey. *arXiv preprint arXiv:2109.13157*, 2021.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489, 2016.

Nong Ye and Qiang Chen. An anomaly detection technique based on a chi-square statistic for detecting intrusions into information systems. *Quality and reliability engineering international*, 17(2):105–112, 2001.

Dit-Yan Yeung and Calvin Chow. Parzen-window network intrusion detectors. In *2002 International Conference on Pattern Recognition*, volume 4, pages 385–388. IEEE, 2002.

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.