

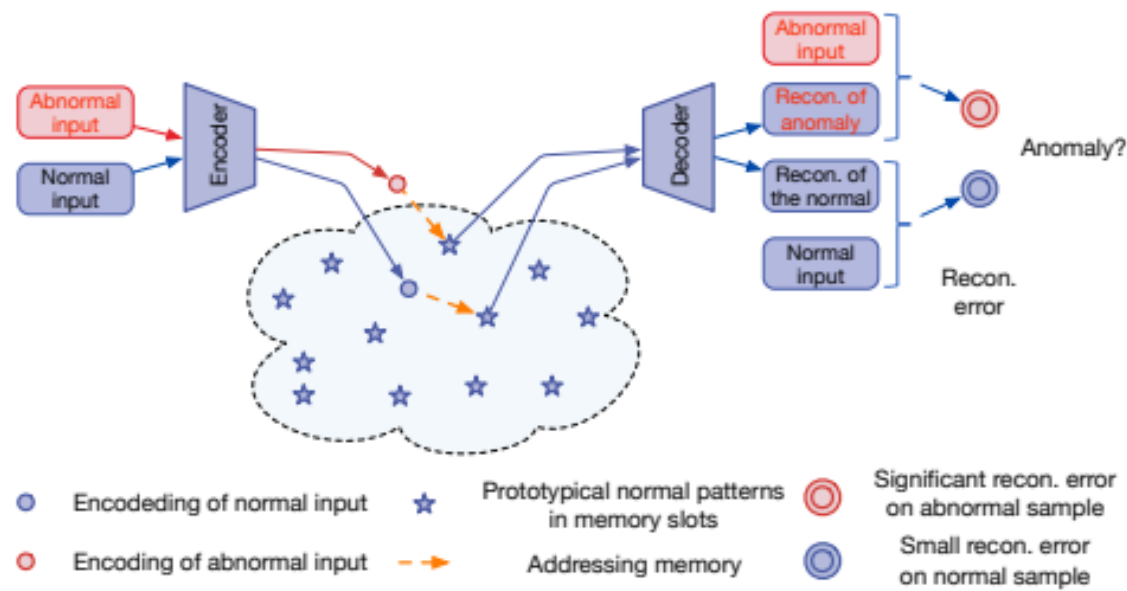
Memorizing Normality to Detect Anomaly: Memory-augmented Deep Autoencoder for Unsupervised Anomaly Detection (MemAE)

≡ Author	Dong Gong ¹ , Lingqiao Liu ¹ , Vuong Le ² , Budhaditya Saha ²
📅 read date	@2022년 3월 29일
▼ Journal	IEEE
📎 PDF	[2019 ICCV] Memorizing Normality to Detect Anomaly: Memory-Augmented Deep Autoencoder for Unsupervised Anomaly Detection.pdf
≡ Published Date	2019
≡ detail	- AE의 한계점 : 종종 이상치 재구성이 잘 됨 - 제안론 구조 : 인코더-메모리모듈(hard shrinkage operator)-디코더 1. 인코더 : 인풋을 인코딩하여 쿼리로 사용 2. 메모리 모듈 1) 표준 정상 패턴을 기록 2) Hard Shrinkage(유사도가 임계치보다 낮은 경우 0으로 만들) → 메모리 효율성 & 이상치의 희소성 ↑ 3. 디코더 : 찾은 유사한 정상 패턴을 통해 재구성 - 결론 : 재구성 에러가 높은 샘플 = 이상치
≡ keyword	Anomaly Detection Unsupervised
🔗 link	https://arxiv.org/pdf/1904.02639.pdf
▼ status	Finished!

Abstract

- Deep autoencoder는 광범위하게 AD에 사용되고 있다. 정상치에 대한 훈련을 통해 AE는 정상치 대비 이상치에 대해 높은 reconstruction error를 생성할 것으로 예상 된다.
- 때때로 AE가 너무 잘 일반화 되어 이상치를 잘 재구성할 수 있어 이상치 검출 능력이 관찰 된다.
→ 이러한 문제 완화를 위해 memory module과 함께 AE를 argument하고, 개선된 AE↓를 개발함
- MemAE(memory-augmented autoencoder) 실행 순서
 1. 입력 단계에서는 MemAE는 인코더로부터 인코딩을 얻고 이를 query로서 사용하여 reconstruction(재구성)에 가장 적합한 memory items(메모리 항목들)을 가져온다.
 2. 훈련 단계에서는 memory contents는 업데이트 되어 정상 데이터의 표준 요소(prototypical elements)를 나타내도록 권장된다.
 3. 테스트 단계에서는 학습된 메모리는 고정되고, 재구성은 정상 데이터의 선택된 메모리 레코드 몇 개로부터 얻어진다. 따라서 재구성은 정상 샘플에 가까운 경향이 있다.
→ 따라서 AD의 reconstructed errors는 AD를 위해 강력해진다.
- MemAE는 데이터 유형에 대한 가정이 없으므로, 다양한 task에 일반적으로 적용된다.
→ 다양한 데이터셋의 실험에서 제안된 MemAE의 높은 효과와 우수한 일반화를 입증한다.
(실험 대상 = 이미지 / 비디오 / 사이버 보안)

1. Introduction



• <Fig 1 설명>

- 일반 샘플로만 훈련 한 후, MemAE의 메모리는 표준의 정상 패턴을 기록한다.
- 이상치 입력이 있는 경우 MemAE는 reconstruction을 위해 메모리 내에서 가장 관련성이 높은 정상 패턴을 검색하여 비정상적인 입력과 큰 차이가 있는 출력을 생성 함.

• AE 이용한 AD의 한계점

- AD는 video 감독분야에서 필수적 task인데, unsupervised anomaly detection는 lack of human supervision으로 어렵다.
- 특히, 고차원 데이터를 모델링 하는 것은 어렵기로 악명 높기에, 고차원 장소(ex. 비디오)안에 데이터가 있을 때 문제가 더 어렵다.

• Deep AE : 비지도 환경의 고차원 데이터를 모델링 하는데 강력한 도구이다.

- Deep AE의 구성 요소 = 인코더 + 디코더
 - 인코더 = 입력으로부터 압축된 인코딩을 취득하는 인코더
 - 디코더 = 인코딩으로부터 데이터를 재구성할 수 있는 디코더
- 인코딩은 본질적으로 고차원 데이터의 전형적인 패턴을 추출하도록 네트워크를 강제하는 정보 병목현상(information bottleneck)으로 작동한다.
- AD의 맥락에서, AE는 일반적으로 정상치의 reconstruction error를 줄이기 위해 훈련되고 이후 이상치의 지표로서 reconstruction error를 사용한다.
 - 일반적으로 정상 데이터는 훈련 데이터에 가깝기 때문에 reconstruction error가 낮지만 비정상 입력은 높은 reconstruction error가 커진다고 가정 된다.
 - 하지만 종종 AE가 이상치 입력을 잘 재구성 하도록 “일반화(generalize)”할 수 있다.
 - 이상치에 대한 훈련 샘플들이 없고 재구성 동작을 예측할 수 없기에 이상치의 reconstruction error가 높다는 가정은 다소 의심스러울 수 있다.
 - 만약 일부 이상치가 정상 훈련 데이터와 일반적인 패턴의 구성을 공유하거나 디코더가 일부 비정상 인코딩을 잘 디코딩 하기 위해 “너무 강한” 경우, AE는 이상치를 잘 재구성할 가능성이 높다.

• MemAE = AE의 문제점을 완화하기 위해 제안됨

- 입력이 주어지면 MemAE는 디코더에 직접 인코딩을 공급하지 않고 메모리 내에서 가장 관련성이 높은 항목을 검색하기 위한 쿼리로 사용한다. → 이후 항목들은 집약되어 디코더에 전달
 - 구체적으로는 어텐션 기반의 memory addressing(3. 참고)의 을 이용하여 위의 프로세스를 실현한다.

• a differentiable hard shrinkage operator 사용을 추가적으로 제안함

- memory addressing weight의 희소성을 유도하기 위해. operator는 메모리 항목이 핏쳐 공간에서 쿼리에 비슷하도록 암묵적으로 장려한다.
 - implicitly encourage the memory items to be close to the query in the feature space??

- 훈련 단계
 - 메모리 내용을 인코더 & 디코더와 같이 업데이트 한다.
 - sparse addressing 전략 때문에 MemAE 모델은 제한된 수의 memory slots을 최적으로 효율적으로 사용하도록 권장되며, 메모리는 일반 훈련 데이터에 표준 정상 패턴을 기록하여 낮은 평균 reconstruction error를 얻는다.
- 테스트 단계
 - 학습된 메모리 내용을 고정하고, 입력 인코딩의 근방으로서 선택된 소수의 정상 메모리 항목을 사용하여 재구성을 얻는다.
 - 재구성은 메모리에서 정상 패턴을 얻기 때문에, 정상 데이터에 가까워 질 수 있다.
 - 입력이 정상 데이터와 유사하지 않은 경우, 즉 이상일 경우 재구성 오류가 강조되는 경향이 있다.
- distribution
 - 데이터 유형에 대한 가정이 없으므로 일반적으로 다양한 과제를 해결하는데 적용될 수 있고
 - 광범위한 실험을 통해 MemAE의 우수한 일반화와 높은 효과를 입증함

2. Related Work

- Anomaly detection
 - : 비지도 AD에서는 정상만 훈련 데이터로 사용할 수 있다.
 - one-class classification methods
 - : 정상 샘플들을 둘러싼 판별하는 초평면(discriminative hyperplane)을 학습하려 함
 - one-class SVM / Deep one-class classification
 - Unsupervised clustering methods
 - : 이상치를 식별하기 위해 정상치의 상세한 윤곽(profile)을 구축하기 위해 적용되었음
 - k-means method / Gaussian Mixture Models (GMM)
 - 이러한 방법들은 고차원 데이터를 처리할 때 성능이 최적화 되지 않는다.
 - Reconstruction-based methods : DAGMM
 - : 정상 데이터에 한하여 모델이 학습됨으로서 이상치는 정확하게 표현 및 재구성 될 수 없다는 가정에 의존한다.
 - 이외 방법들 : PCA methods / sparse representation (ex. Spatio-temporal AE for video AD)
 - : 정상 패턴의 표현을 학습하기 위해 사용되어 왔다.
 - sparse representation은 사전(dictionary)과 정상 데이터의 희박한 표현(sparse representation)을 공동으로 학습한다.
 - 제한된 핏처 표현은 퍼포먼스에 한계가 있다.
 - Deep AE for anomaly detection (최신 연구) : [43, 46, 48, 6]
 - [43] Deep structured energy based models for AD : 훈련 샘플 모델링에 사용 됨
 - [46] Anomaly detection with robust deep **autoencoders**
 - [48] DAGMM : Deep **AE**에서 인코딩된 핏처와 재구성 에러를 공동으로 모델링
 - [6] Abnormal event detection in videos using spatiotemporal **autoencoder**
 - 재구성 기반 방법들이 많은 성과를 거두었지만, 성능은 laten space의 충분히 설계되지 않은 표현에 의해서 제한된다.
 - video anomaly detection
 - [24, 44, 11, 21]

- [15] MPPCA : 광학적 흐름 핏처를 모델링 하기 위해 확률론적 PCA의 혼합물 사용
- [27] MDT : 동적 텍스처를 혼합하여 비디오를 모델링
- [25] an efficient sparse coding-based method with multiple dictionaries
 - : 다중 사전 사용하는 효율적인 sparse coding-based model
 - sparse coding 이란? 조각 영상을 여러 개의 기저 벡터 선형 결합으로 표현하는 것
- [44] update the dictionary in an online manner : 온라인 방식으로 딕셔너리 업데이트

◦ spatial and temporal domain 정보 사용을 위한 DL 기반 methods

- [11] convolutional AE의 reconstruction error에 따른 이상치 감지
- [45] 3D convolution based reconstruction and prediction
- [26] stacked RNN을 통한 sparse coefficients 반복적 업데이트로 비디오 이상 감지
- [24] frame prediction network by incorporating(통합) different techniques including gradient loss, optical flow(광학적 흐름), and adversarial training.
 - 모델이 이상치에서 큰 재구성 오류를 유도하도록 하는데, 이를 신뢰할 수 있는 방법이 없다.

• Memory networks

: 메모리 증강 네트워크는 다양한 문제를 해결하는데 점점 더 많은 흥미를 끌고 있다.

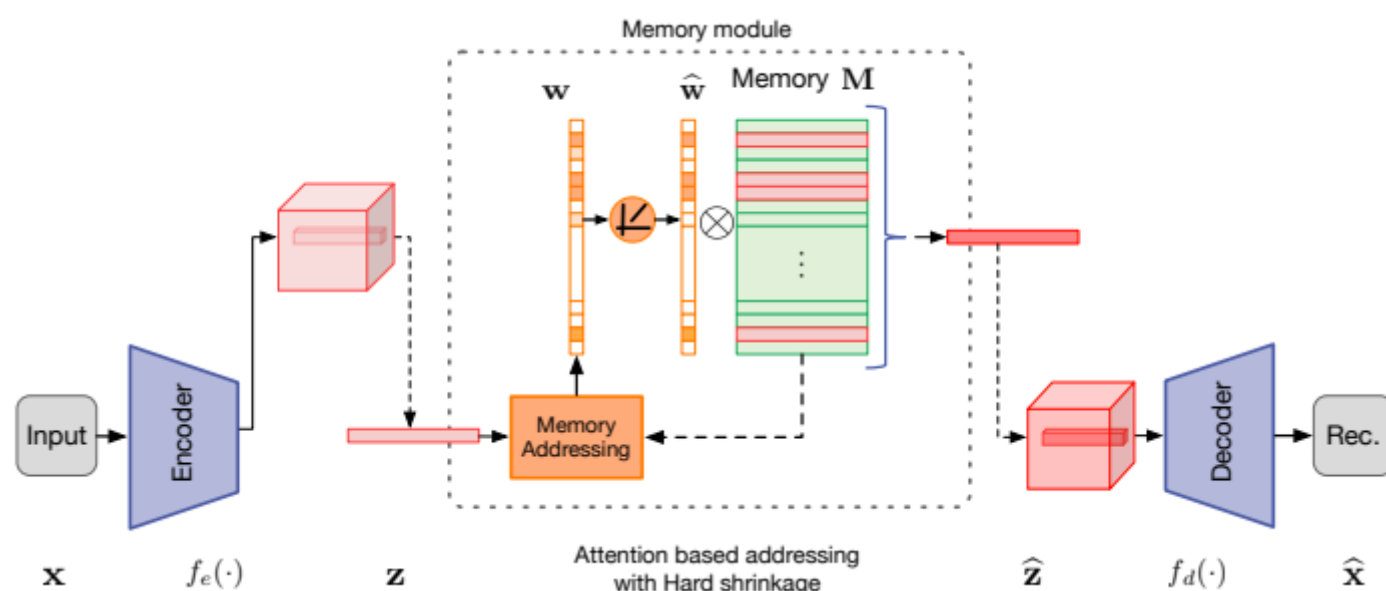
- [10] NN의 능력을 증가 시키기 위해 외부 메모리를 사용
- [33] 메모리가 정보를 안정적으로 기록할 수 있게 고려됨
- [16,22] 외부의 메모리는 모드 축소 문제를 회피하고 상세한 데이터 구조를 보존하기 위해 multi-modal data 생성에도 사용되고 있다. ?? 설명 필요
 - [16] Memorization precedes generation: Learning unsupervised gans with memory networks.
 - [22] Learning to generate with memory

3. Memory-augmented Autoencoder

1. Overview

- MemAE 모델의 주요 3개 요소

: encoder(입력 인코딩 및 쿼리 생성), decoder(재구성), memory module(메모리와 메모리 주소 지정 연산자)



<Fig 2 설명>

- memory addressing unit은 인코딩 z를 soft addressing weights를 얻기 위해 쿼리로 사용
- memory slots은 전체 인코딩 또는 인코딩의 한 개 픽셀 핏처 모델에 사용될 수 있다.
- \hat{w} 는 hard shrinkage operation 이후 정규화 된다.

2. Encoder and Decoder

- 인코딩된 표현은 메모리 내의 관련 항목을 검색하기 위한 쿼리로 사용 가능함
- 제안 된 방법에서는 인코더 query generator로 볼 수 있다.
- 디코더는 검색된 메모리를 입력으로 받아 샘플을 재 구성하도록 훈련 됨
- 인코더 = $f_e(\cdot) : X \rightarrow Z$, 디코더 = $f_d(\cdot) : \hat{Z} \rightarrow X$

$$\mathbf{z} = f_e(\mathbf{x}; \theta_e),$$

$$\hat{\mathbf{x}} = f_d(\hat{\mathbf{z}}; \theta_d),$$

- θ_e, θ_d = 각각 인코더 디코더의 파라미터들
- \mathbf{z} 를 사용하여 관련 메모리 항목을 검색하고, $\hat{\mathbf{z}}$ 는 관련 항목을 통해 얻어진다.
 - 일반 AE는 $\hat{\mathbf{z}} = \mathbf{z}$ 이지만, 제안에선 인코더와 디코더 구조에 의존하지 않고 다양한 응용을 위해 특별히 선택할 수 있다.
- 평가에선 \mathbf{x} 를 준뒤, ℓ_2 -norm 기반 MSE를 사용. 재구성 품질을 측정하기 위해

3. Memory Module with Attention-based Sparse Addressing

: 메모리 모듈 = 표준 인코딩된 패턴을 기록하는 메모리 + 메모리에 접근하기 위한 기반 주소 지정 연산자로 구성됨

- Memory-based Representation (메모리 기반 표현)

$$\hat{\mathbf{z}} = \mathbf{w}\mathbf{M} = \sum_{i=1}^N w_i \mathbf{m}_i,$$

- \mathbf{w} = soft addressing vector(weight vector=쿼리인 \mathbf{z} 를 통해 계산 됨)
- \mathbf{M} = memory matrix, N = 최대 메모리 용량
- Attention for Memory Addressing (메모리 주소 지정을 위한 Attention)
 - memory \mathbf{M} 은 훈련 동안 표준 정상 패턴으로 명시적 기록되도록 설계되었다.
 - 메모리를 쿼리 \mathbf{z} 와 메모리 항목의 유사함에 기반한 어텐션 웨이트 \mathbf{w} 를 계산하는 주소 지정 스키마인 연산 기억 장치로 정의한다.

$$w_i = \frac{\exp(d(\mathbf{z}, \mathbf{m}_i))}{\sum_{j=1}^N \exp(d(\mathbf{z}, \mathbf{m}_j))},$$

- 소프트맥스 연산을 통해 각 w_i (=유사도 책정한 attention weight)를 계산함
- $d(\cdot, \cdot)$ = 유사도 측정(similarity measurement) as cosine similarity

$$d(\mathbf{z}, \mathbf{m}_i) = \frac{\mathbf{z}\mathbf{m}_i^T}{\|\mathbf{z}\|\|\mathbf{m}_i\|}.$$

- 가장 유사한 메모리 항목을 검색하여 표현 $\hat{\mathbf{z}}$ 를 얻음
- 한정된 메모리 사이즈와 희박한 주소 지정 기술 때문에 매번 소수의 메모리 항목만 주소를 지정할 수 있다.
- training 단계
 - 디코더는 매우 적은 수의 지정된 메모리 항목을 사용하여 재구성을 수행하도록 제한됨으로 메모리 항목을 효율적으로 관리해야 함
 - 재구성 감시는 입력 정상 패턴에서 가장 대표적인 표준 패턴을 메모리에 기록
= 각 메모리 슬롯이 훈련 데이터를 통해 표준 정상 패턴을 기록하고 있다.

- testing 단계
 - 훈련된 메모리가 주어지면, 재구성을 위해 메모리의 정상 패턴 만 검색될 수 있다.
 - 정상치는 잘 재구성 되고, 이상치의 인코딩은 검색된 정상 패턴으로 대체되어 상당한 재구성 에러가 발생한다.

- Hard Shrinkage for Sparse Addressing

- 왜 쓰는지??
 - 내 생각 = 일부 이상치는 잘 재구성 될 수 있기 때문에 w 의 희귀성을 높이기 위해 적용되는데, 유사도 > 임계치인 경우 유사도 값을 출력하고, 유사도 < 임계치인 경우 0으로 만들어서 희소성 있게 만든다. 이것은 ReLU 구조와 같다.
- 메모리에서 제한된 수의 정상 패턴으로 재구성을 수행하면 이상치를 큰 재구성 오류를 유도하는데 도움이 됨 → 어텐션 기반 주소 지정은 자연스럽게 접근하는 경향이 있다.
- 일부 이상치는 작은 요소를 많이 포함하는 밀도 w 를 통해 메모리 항목의 복잡한 조합을 사용하여 잘 재구성 될 수 있다.
→ 해결하기 위해 w 의 희소성을 증가시키기 위해 hard shrinkage operation을 적용

$$\hat{w}_i = h(w_i; \lambda) = \begin{cases} w_i, & \text{if } w_i > \lambda, \\ 0, & \text{otherwise,} \end{cases}$$

- \hat{w}_i = shrinkage 후 메모리 주소 지정 weight vector인 \hat{w} 의 i 번째 순서를 나타낸다.
- λ = shrinkage(수축) threshold(임계 값)
- hard shrinkage operation 재 작성 : ReLU 함수 구조

$$\hat{w}_i = \frac{\max(w_i - \lambda, 0) \cdot w_i}{|w_i - \lambda| + \epsilon},$$

- $\max(\cdot, 0)$ = ReLU 활성화 함수, ϵ = 매우 작은 양의 scalar

$$\hat{w}_i = \hat{w}_i / \|\hat{\mathbf{w}}\|_1, \forall i.$$

- shrinkage 후 \hat{w} 를 위에와 같이 재-정규화 함 → 잠재 표현 $\hat{z} = \hat{\mathbf{w}}\mathbf{M}$ 를 얻음
- 정리 : sparse addressing은 모델이 더 적은 수의 관련 메모리 항목을 사용하여 예시를 표현하도록 장려하여 메모리 내에서 더 많은 정보 표현을 학습하도록 유도한다,
→ 희소성을 장려하여 비정상적인 샘플이 dense addressing weights로 충분히 재구성 될 수 있다는 문제가 완화될 것

4. Training

- minimize reconstruction error

$$R(\mathbf{x}^t, \hat{\mathbf{x}}^t) = \|\mathbf{x}^t - \hat{\mathbf{x}}^t\|_2^2,$$

- $\hat{\mathbf{x}}^t$ = 각 훈련 샘플 \mathbf{x}^t 에 상응하는 재구성된 샘플
- ℓ_2 -norm은 재구성 오류 측정에 사용 됨
- $\hat{\mathbf{w}}^t$ = 각 샘플 \mathbf{x}^t 를 위한 memory addressing weights
- entropy loss : 학습 중 $\hat{\mathbf{w}}$ 에 대한 sparsity regularizer(희소성 정규화기)를 최소화

$$E(\hat{\mathbf{w}}^t) = \sum_{i=1}^T -\hat{w}_i \cdot \log(\hat{w}_i).$$

- Total loss object function for MemAE

$$L(\theta_e, \theta_d, \mathbf{M}) = \frac{1}{T} \sum_{t=1}^T (R(\mathbf{x}^t, \hat{\mathbf{x}}^t) + \alpha E(\hat{\mathbf{w}}^t)),$$

- α = hyper-parameter in training, T = 샘플의 크기
- 훈련 동안 메모리 M은 역전파 및 SGD를 통해 갱신 됨

4. Experiments

- MemA 평가 챕터로 3개의 task와 5개의 데이터셋으로 실험함

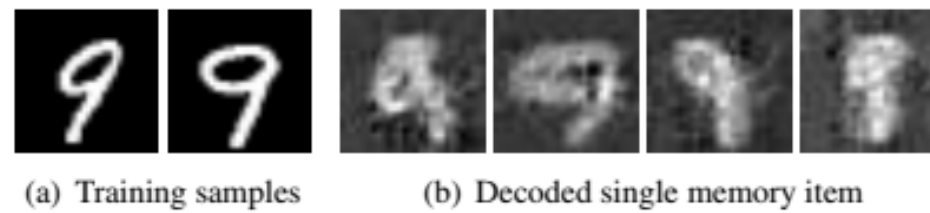
1. Experiments on Image Data : to detect outliers image

Table 1. Experimental results on image data. Average AUC values on 10 anomaly detection datasets sampled from MNIST and CIFAR-10 are shown.

Dataset	MNIST	CIFAR-10
OC-SVM [35]	0.9499	0.5619
KDE	0.8116	0.5756
VAE [18]	0.9643	0.5725
PixCNN [38]	0.6141	0.5450
DSEBM [43]	0.9554	0.5725
AE	0.9619	0.5706
MemAE-nonSpar	0.9725	0.6058
MemAE	0.9751	0.6088

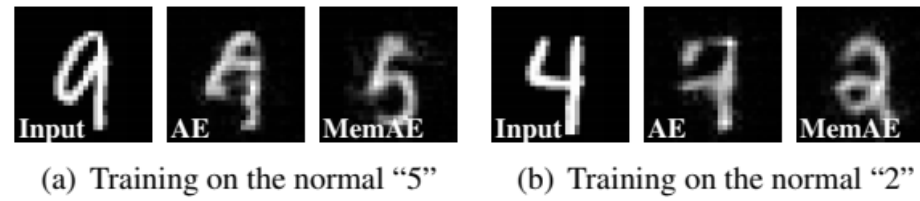
- < Table 1 설명 >
 - Anomaly Detection 평균 AUC 유사한 용량을 갖는 모델 중 MemAE는 우수한 성능을 달성 → 제안된 메모리 모듈의 효과를 입증
 - AE : 메모리 모듈이 없는 / MemAE-nonSpar : sparse shrinkage & entropy loss가 없는
 - AE, MemAE-non-Spar, MemAE는 동일한 인코더와 디코딩을 사용해서 동일한 용량을 공유 함
- datasets = MNIST [20] and CIFAR-10 [19] (10개의 정상 클래스와 1개의 이상 클래스)
- 정상 데이터는 2:1로 훈련:테스트 데이터로 분할 됨. 이상치는 30%로 통제되며, 10%는 검증을 위해 남겨진다.
- 인코더와 디코더를 CNN으로 구현한다. (MNIST 기준 ↓, CIFAR은 더 복잡)
 - Conv2(k, s, c) = 각 kerner size, stride size, num of channel
 - 인코더 = Conv2(3, 2, 32)- Conv2(3, 2, 16)-Conv2(3, 3, 8)
 - 디코더 = Dconv2(3, 3, 16)-Dconv2(3, 2, 32)-Dconv2(3, 2, 1)
 - Dconv2 = 2D Deconvolution ?
 - : convolution layer는 feaure map의 크기 ↓, Deconvolution은 feature map의 크기 ↑
 - cnn의 최종 결과물을 되돌려 input과 같은 사이즈를 만들어내고 싶을 때 사용 함
 - test에서는 이상치 식별의 기준으로 reconstruction error를 0~1 사이로 스케일링함
- VAE ?

VAE의 Encoder 부분을 통해서 mean와 std를 학습하고, 이 mean과 std를 통해 Gaussian distribution에서 randomly chosen value를 선택하고 이 value를 decoder에 전달함으로써 동일하지 않은 unique한 (하지만 유사한) output을 생성할 수 있다. 즉, 표준 정규 분포에 가까워지도록 학습함에 따라 AE보다 분포가 고른 형태를 보이게 된다.
- Visualizing How the Memory Works
 - What the memory learns.



- 메모리 슬롯을 랜덤으로 샘플링 해 디코딩을 실시함. 오른쪽과 같이 디코딩 된 결과는 흐릿하고 노이즈가 발생하지만, 그럼에도 불구하고 일반 교육 샘플의 다양한 표준의 패턴을 기록한다.

- How memory augments reconstruction.



- 왼쪽은 모델이 5로 훈련되었지만, 9의 인풋을 넣은 결과이고, 오른쪽은 2로 훈련이 되었지만 4를 입력으로 넣은 결과이다. MemAE는 입력과는 상당히 다른 결과를 보인다.
- 훈련된 메모리는 정상적인 표준 패턴만 기록하기 때문에, 비정상인 입력 9가 주어졌을 때 MemAE는 5를 재구성하여 비정상적인 입력에 대한 상당한 reconstruction error가 발생한다. → 가장 유사한 패턴을 가오기 때문에 재구성된 5는 9와 유사함
- 메모리가 없는 AE 모델은 일부 표현을 locally 더 많이 학습하는 경향이 있어서 이상치도 잘 재구성 될 수 있다.

2. Experiments on Video Anomaly Detection

: 비디오에서 일반적이지 않은 움직이는 패턴과 요소를 판단하는 것을 목표로 함

Table 2. AUC of different methods on video datasets UCSD-Ped2, CUHK Avenue and ShanghaiTech.

Method\Dataset		UCSD-Ped2	CUHK	SH.Tech
Non-Recon.	MPPCA [15]	0.693	-	-
	MPPCA+SFA [27]	0.613	-	-
	MDT [27]	0.829	-	-
	AMDN [41]	0.908	-	-
	Unmasking [37]	0.822	0.806	-
	MT-FRCN [12]	0.922	-	-
	Frame-Pred [26]	0.954	0.849	0.728
Recon.	AE-Conv2D [11]	0.850	0.800	0.609
	AE-Conv3D [45]	0.912	0.771	-
	TSC [26]	0.910	0.806	0.679
	StackRNN [26]	0.922	0.817	0.680
	AE	0.917	0.810	0.697
	MemAE-nonSpar	0.929	0.821	0.688
	MemAE	0.941	0.833	0.712

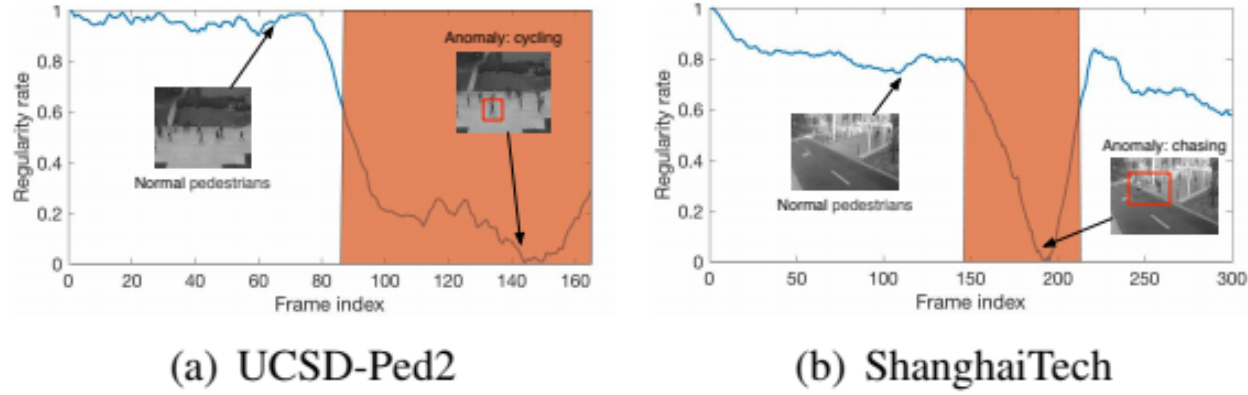
• <Table 2 설명>

- UCSD-Ped2, CUHK Avenue, ShanghaiTech 3개의 데이터 셋을 사용
 - ShanghaiTech 기준 이상치 : 보행자(ex. 차량), 격렬한 움직임(ex. 전투 및 추격)을 제외한 물체는 이상치 취급
- Non-Recon. = non-reconstruction techniques
- 비디오 데이터 셋의 AUC 값을 비교했을 때, 제안 모델이 높은 점수를 기록한다.
- 비디오 시간적 정보를 보존하기 위해 3D CNN을 이용하여 인코더와 디코더를 구현하여 비디오의 공간-시간 핏처를 추출함
 - 네트워크 입력 : 16개의 인접 프레임을 그레이 스케일로 쌓은 직육면체
 - 인코더 = Conv3(3, 2, 96)- Conv3(3, 2, 128)-Conv3(3, 2, 256)-Conv3(3, 2, 256)
 - 디코더 = Dconv3(3, 2, 256)-Dconv3(3, 2, 128)-Dconv3(3, 2, 96)- Dconv3(3, 2, 1)

- 각 메모리 슬롯이 비디오 클립 하위 영역에 해당하는 feature map의 1 pixel에 feature를 기록하도록 함 → 메모리는 2000x256 크기의 행렬
- test에서는 각 프레임을 중심으로 한 정육면체의 reconstruction error에 의해 평가 됨

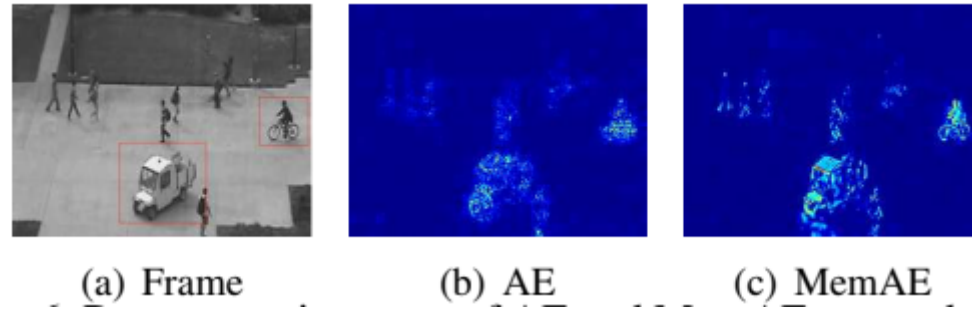
$$p_u = 1 - \frac{e_u - \min_u(e_u)}{\max_u(e_u) - \min_u(e_u)},$$

- errors 범위를 0~1로 정규화하여 u -th번째 프레임의 normality score인 p_u 를 얻음
- e_u = 비디오 에피소드에서 u -th번째 프레임의 reconstruction error를 나타냄
- p_u 값이 0에 가까울 경우 이상 프레임일 가능성이 높다는 것을 나타냄



◦ <Fig 5. 설명>

- MemAE로부터 얻은 비디오 프레임의 정상 점수로 점수는 비디오 프레임에서 조금의 이상치 모습이 있을 때 즉시 감소한다.

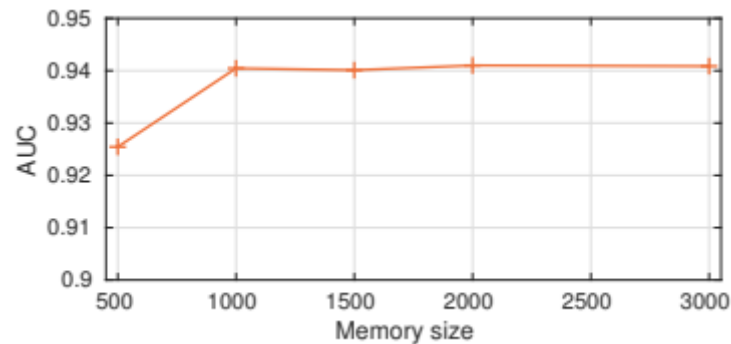


◦ <Fig 6. 설명>

- UCSD-Ped2의 이상치 프레임에 대한 AE, MemAE의 error map
- MemAE의 경우 이상치 부분이 크게 강조되어 있음을 볼 수 있다.

• Robustness to the memory size

: UCSD-Ped2를 사용하여 MemAE의 메모리 사이즈 N에 대한 robustness를 조사한다.



- 다양한 메모리 사이즈에 대한 AUC를 봤을 때, 충분한 메모리 사이즈면 확실한 결과를 얻을 수 있음을 볼 수 있다.

• Running time

- NVIDIA GeForce 1080 Ti로 UCSD-Ped2를 실험했을 때, 0.0262초/frame가 소요됨

→ 이는 이전의 최첨단 딥러닝 기반 방법보다 동등하거나 더 빠르다.

3. Experiments on Cybersecurity Data (사이버 보안 데이터 실험)

Table 3. Experimental results of different methods on the cybersecurity dataset KDDCUP.

Method\Metric	Precision	Recall	F_1
OC-SVM [35]	0.7457	0.8523	0.7954
DCN [42]	0.7696	0.7829	0.7762
DSEBM [43]	0.8619	0.6446	0.7399
DAGMM [48]	0.9297	0.9442	0.9369
AE	0.9328	0.9356	0.9342
MemAE-nonSpar	0.9341	0.9368	0.9355
MemAE	0.9627	0.9655	0.9641

- 제일 좋은 성능을 나타냄

4. Ablation Studies

: 다른 구성 요소를 자세히 조사하기 위해 몇 가지 ablation study 추가

- Study of the Sparsity-inducing Components

Table 4. Ablation studies based on UCSD-Ped2 dataset.

Method	AUC
AE	0.9170
AE- ℓ_1	0.9286
MemAE-nonSpar	0.9293
MemAE w/o Shrinkage	0.9324
MemAE w/o Entropy loss	0.9372
MemAE	0.9410

- memory addressing weights의 희소성을 유도하기 위해 hard thresholding shrinkage & entropy loss를 사용하는데, 각 요소가 중요함을 알 수 있다.
- AE with Sparse Regularization(AE- ℓ_1)를 사용해서 잠재 압축 기능을 최소화하면 기존 AE 대비 좋은 성능을 가진다.

5. Conclusion

- 미래에 addressing weight를 사용하여 이상치 감지하는걸 검토할 예정
- 제안된 메모리 모듈은 일반적이며, 인코더와 디코더 구조에 의존하지 않는다는 점을 고려하여 보다 복잡한 기본 모델에 통합하여 보다 까다로운 응용에 적용할 수 있다.