

논문 2025-3-5 <http://dx.doi.org/10.29056/jsav.2025.09.05>

D²MR: 머신러닝 시스템에서 데이터 특성 기반의 규칙 매핑을 통한 드리프트 탐지 기법 추천 프레임워크

최옥주*†

D²MR: A Framework for Recommending Drift Detection Method via Data Characteristic-based Rule Mapping in Machine Learning System

Okjoo Choi*†

요 약

머신러닝 시스템에서 시간이 지남에 따라 발생하는 데이터 분포 변화로 인한 드리프트는 성능 저하와 모델의 신뢰성 약화를 초래한다. 이를 해결하기 위해 다양한 드리프트 탐지 기법들이 연구되었으나 실제 환경에서는 데이터 특성에 적합한 기법을 선정하는 데 어려움이 존재한다. 본 연구에서는 데이터셋의 메타 특성(라벨 존재 여부, 데이터 유형, 크기, 차원 수, 거리 계산 가능성, 분포 유형)을 기반으로 적합한 탐지 기법을 추천하는 D²MR(Drift Detection Method Recommender) 프레임워크를 제안한다. 제안하는 프레임워크는 데이터 특성과 탐지 기법을 체계적으로 매핑하여 머신러닝 시스템의 드리프트 모니터링에 효과적으로 통합될 수 있다. 본 연구에서는 CIFAR-10 이미지 데이터셋과 UCI 와인품질 데이터셋을 활용한 사례연구를 통해 D²MR의 실효성을 검증하였다.

Abstract

Machine learning systems often suffer from performance degradation and reduced model reliability due to data distribution shifts (drift) that occur over time. Although various drift detection methods have been studied to address this issue, selecting a method that is well-suited to the characteristics of real-world datasets remains challenging. This study proposes the D²MR(Drift Detection Method Recommender) framework, which automatically recommends suitable detection methods based on dataset meta-features such as label availability, data type, size, dimensionality, distance computability, and distribution type. The proposed framework systematically maps data characteristics to detection methods and can be effectively integrated into drift monitoring modules of machine learning systems. Finally, the effectiveness of D²MR is validated through case studies using the CIFAR-10 image dataset and the UCI Wine Quality dataset.

한글키워드 : 데이터 특성, 드리프트, 드리프트 탐지 기법, 머신러닝 시스템

keywords : Data Characteristics, Drift, Drift Detection Method, Machine Learning System

* 배재대학교 AI.소프트웨어공학부

† 교신저자: 최옥주(email: okjoo.choi@pcu.ac.kr)

접수일자: 2025.08.31. 심사완료: 2025.09.12.

게재확정: 2025.09.20.

1. 서론

머신러닝(ML, Machine Learning) 시스템은

이미지 분류, 음성 인식, 재무 예측, 의료 진단 등 광범위한 응용 분야에서 필수가 되었다. 다양한 산업 분야에서 ML시스템에 대한 의존도가 증가함에 따라 ML모델 배포 이후에도 모델의 정확성과 지속적인 성능을 유지하는 것이 점점 더 중요해졌다[1]. 현실 세계에서 ML 시스템이 직면하는 성능 유지를 위한 주요 과제 중 하나가 드리프트 현상이다. 드리프트는 입력 데이터의 통계적 특성이 시간이 지남에 따라 변하여 모델 성능을 저하시키는 현상이다[2]. 예를 들어, 안정적인 시장 데이터를 사용하여 회사의 주가를 예측하도록 학습된 모델은 처음에는 좋은 성능이었으나 시장 변동성이 증가함에 따라 데이터 분포의 변화로 인해 모델의 예측이 부정확해질 수 있다[3]. 이러한 드리프트가 발생하는 ML 시스템의 성능 저하는 지속적인 신뢰성이 매우 중요한 운영 환경에서 심각한 문제가 야기되고 있다[4].

데이터의 통계적 특성이 변하거나 학습된 초기 ML 모델의 입력과 출력간의 관계가 변화하는 등 다양한 유형으로 발생하는 드리프트 현상을 해결하기 위해 많은 드리프트 탐지 기법들이 연구되고 관련 알고리즘이 개발되었다. 그러나 이러한 드리프트 탐지 기법들을 현실적으로 적용하고자 할 때 어떤 기준으로 가장 적합한 탐지 기법을 선정해야 하는지 판단하기가 쉽지 않다. 기존 연구에서는 발생하는 드리프트 유형을 미리 결정하고 임의의 기법을 선정하여 사용하거나 동시에 여러 기법을 실험한 후 성능 좋은 드리프트 탐지 기법을 적용하는 선정 기법이 체계적이지 못하다. 또한, 대부분의 기존 연구는 탐지 기법의 비교나 성능 평가에 초점을 맞추고 있어서 어떤 상황에서 어떤 기법을 선택해야 하는지에 대한 구체적 기준을 제시하고 있지 않다.

실제 운영하는 ML 시스템에서 시간이 지남에 따른 변화를 모니터링하기 위해 제안된 알고리즘은 데이터 특성 및 드리프트 유형에 따라 장점과

한계를 가지고 있다. 따라서 정해진 시나리오에 적합한 탐지 기법을 선택하는 것은 동적인 환경에서는 중요한 과제로 남아 있다.

본 연구에서는 이러한 문제를 해결하기 위해 데이터의 메타 특성의 분류와 기준을 제시하고 이를 기반으로 드리프트 탐지 기법을 체계적으로 매핑한다. 이를 기반으로 특정 데이터셋에 적합한 드리프트 탐지 기법을 추천하는 프레임워크를 제안한다. 서로 다른 CIFAR-10 이미지 데이터셋과 UCI 와인 데이터셋을 활용한 사례연구를 통해 제안한 프레임워크를 검증하고 ML 생명주기와의 통합으로 드리프트 모니터링을 보장하고 ML 시스템의 성능 유지의 가능성을 제시한다.

2. 관련 연구

2.1 드리프트 유형

1) 데이터 드리프트

데이터 드리프트(Data Drift)는 ML 모델을 학습할 때와는 달리 입력 데이터(x)의 특징 분포가 변화하여 발생한다[5].

$$P_t(x) \neq P_{t+\Delta}(x) \quad (\text{식-1})$$

- $P_t(x)$: 시간 t에서의 입력 데이터 분포
- $P_{t+\Delta}(x)$: 시간 t+Δ에서의 입력 데이터 분포

예를 들어, 연령과 소득을 기준으로 고객이 제품을 구매할 가능성을 예측하도록 학습된 ML 모델에서 시간이 지남에 따라 고객의 연령 및 소득 분포가 크게 변하는 경우 모델은 더 이상 구매 가능성을 정확하게 예측하지 못하게 된다.

2) 개념 드리프트

개념 드리프트(Concept Drift)는 개념 드리프트는 ML 모델이 학습한 입력 데이터(x)와 목표 결과(y) 사이의 관계 변화로 시간 경과에 따른 조건부 분포($P(y|x)$)의 변화가 발생한다.

$$P_t(y | x) \neq P_{t+\Delta t}(y | x) \quad (\text{식-2})$$

- $P_t(y | x)$: 시간 t 에서의 입출력 관계
 - $P_{t+\Delta t}(y | x)$: 시간 $t+\Delta t$ 에서의 입출력 관계

이는 ML 모델의 예측 결과가 달라지는 것을 의미한다. 예를 들어, 이메일 내용을 기반으로 스팸을 탐지하는 ML 모델을 학습했는데 시간이 지나 스팸메일의 유형이 변경되면 ML 모델은 더 이상 스팸을 정확하게 탐지할 수 없게 된다. 개념 드리프트는 발생 형태에 따라 급격한 드리프트(sudden drift), 점진적 드리프트(gradual drift), 누적적 드리프트(incremental drift), 반복적 드리프트(recurring drift)로 상세 분류한다[6].

급격한 드리프트는 특정 시점에서 모델의 예측 구조가 급격히 바뀌는 현상이고 점진적 드리프트는 변화가 일정 기간 동안 천천히 진행되어 과거와 새로운 개념이 공존하는 현상이다. 누적적 드리프트는 ML모델의 예측값이 연속적으로 천천히 변화하여 기존의 예측값은 점점 사라지고 새로운 예측값만 발생하는 현상이다. 반복적 드리프트는 과거에 나타난 예측값이 특정 주기로 반복해서 나타나는 현상이다.

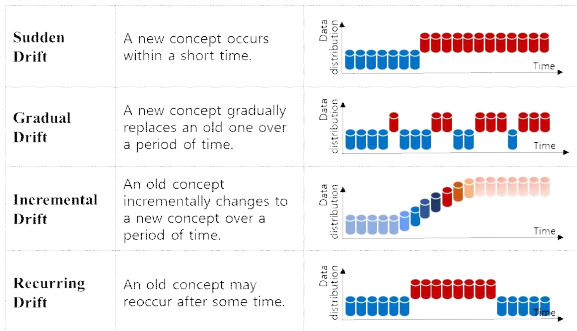


그림 1. 개념 드리프트 유형
 Fig 1. Concept Drift Type

3) 사전확률 드리프트

사전확률 드리프트(Prior Probability Drift)는 출력된 클래스 사전확률이 시간에 따라 변하는

현상이다[7]. 입력과 출력간의 조건부 확률 $P(y | x)$ 는 변하지 않고 클래스 분포 자체가 달라지는 현상이다.

$$P_t(y) \neq P_{t+\Delta t}(y) \quad (\text{식-3})$$

- $P_t(y)$: 시간 t 에서의 클래스 분포
 - $P_{t+\Delta t}(y)$: 시간 $t+\Delta t$ 에서의 클래스 분포

예를 들어, 스팸이 전체 메일의 5%였던 것이 시간이 지나면서 30%까지 증가하는 경우이다.

2.2 드리프트 탐지 기법

2.2.1 통계 기반 탐지 기법

통계 기반 탐지 기법은 통계 지표를 사용하여 학습 데이터와 현재(운영) 데이터 사이의 분포변화를 측정한다. 이러한 변화는 시스템 운영 중에 나타나므로 드리프트 탐지를 위한 통계 기법은 시간 경과에 따라 기준 통계량과 유의미한 편차를 보이는지 평가한다.

- PSI

PSI(Population Stability Index)는 학습 데이터셋과 운영 데이터셋이 동일한 분포인지 평가하는 지표이다[8]. 새롭게 입력되는 운영 데이터셋에서 변수의 분포를 학습 데이터셋과 비교하여 변수 분포의 변화를 측정한다. PSI 값이 낮을수록 두 데이터셋의 분포가 더 유사하고 안정적이다. PSI 값이 높을수록 두 데이터셋의 분포가 다르고 불안정함을 의미한다.

- KL Divergence

KL(Kullback-Leibler) Divergence는 예측값의 분포가 참(true value)인 분포와 얼마나 다른지를 측정하는 값이다[9]. KL Divergence 결과는 항상 0 이상인 값을 반환한다. KL Divergence 결과값이 운영 데이터의 분포와 학습 데이터가 특정 임계값(threshold)을 초과하면 드리프트가 발생한 것으로 판단한다.

- JS Divergence

JS(Jensen-Shannon) Divergence은 두 데이터셋의 확률분포 간의 유사도를 측정한다[10]. 이 유사도는 정규화된 값으로 0과 1 사이의 값으로 반환된다. KS Divergence와 대칭화된 버전의 지표로 자연어처리나 클러스터링에서 사용된다.

- KS Test

KS(Kolmogorov-Smirnov) Test는 두 데이터셋이 동일한 분포 인지를 따르는지 검정하기 위해 분포를 비교하는 데 사용되는 비모수적 통계 검정(non-parametric test)이다[2][11]. 비교되는 두 데이터셋의 누적 분포 함수간의 최대 차이를 측정한다. KS Test는 두 데이터셋이 동일한 분포에서 유래했는지 여부를 판별한다. KS Test 결과 p-value가 0.05 미만이면 두 데이터셋의 분포가 다르다는 것을 나타내며 잠재적으로 드리프트가 탐지될 가능성이 있다고 판단한다.

- MMD

MMD(Maximum Mean Discrepancy)는 두 확률분포 간의 차이를 정량화하는 커널 기반 통계적 거리 측정 기법이다[12]. 학습 데이터와 새롭게 입력되는 운영 데이터의 평균 임베딩 벡터 차이를 계산하여 분포 차이를 감지한다.

- FID

FID(Fréchet Inception Distance)는 이미지 분포의 유사성을 측정하기 위한 통계 지표이다[13][14]. 두 이미지 데이터셋(생성 이미지와 실제 이미지)에서 추출한 피처(feature)의 분포 평균과 공분산을 구한 후 이들 간의 Fréchet 거리를 계산한다. Fréchet 거리가 0에 가까울수록 생성 이미지 분포가 실제와 유사하다고 판단한다. FID 기법은 주로 GAN(Generative Adversarial Network)의 생성 이미지 품질을 평가하기 위해 사용되는 지표로서 시간 흐름에 따른 이미지 데이터 분포 변화 감지에 활용한다.

- LPIPS

LPIPS(Learned Perceptual Image Patch

Similarity)는 이미지의 시각적 품질 차이를 정량화하기 위해 학습된 신경망 기반의 거리 측정 기법이다[15]. 각 이미지의 임베딩 차이를 여러 계층에서 측정하여 사람이 인지하는 유사도와 유사한 값을 제공한다.

2.2.2 스트리밍 기반 탐지 기법

운영 데이터셋의 통계 특징인 평균, 분산, 왜도, 첨도와 같은 기본적인 통계 측정값을 시간 경과에 따라 모니터링하면서 훈련 데이터셋의 기준 통계와의 유의미한 차이가 있으면 드리프트가 발생한다고 판단한다.

- DDM

DDM(Drift Detection Method)은 오류율과 해당 표준 편차를 분석하여 스트리밍 데이터셋의 드리프트 유무를 탐지한다[16]. 운영 데이터셋 i 번째 피처에 대해 잘못된 예측을 할 확률인 오류율 p_i 와 표준 편차 $s_i = \sqrt{p_i \times (1 - p_i) / i}$ 를 정의한다. 운영 데이터셋의 오류율이 증가하면 데이터 분포의 변경으로 인해 드리프트가 발생했다고 판단한다.

- EDDM

EDDM(Early DDM)은 DDM과 유사하지만 오류율의 변화 감시가 아닌 두 개의 연속된 오류 사이의 거리를 평가한다[17]. 연속된 오류 사이의 평균 거리와 표준편차를 추적하여 오류 사이의 거리가 감소하면 드리프트가 발생했다고 판단한다. EDDM은 점진적인 개념 드리프트를 탐지하는데 적합하고 DDM은 급격한 개념 드리프트를 탐지하는데 적합하다.

- HDDM

HDDM(Hopping Drift Detection Method)은 온라인 데이터 스트림에서 드리프트를 탐지하기 위한 기법이다[18]. 주된 특징은 “hopping” 윈도우 전략으로 다양한 길이의 윈도우에서의 패턴 변화를 기반으로 드리프트를 탐지한다. 분류기의

에러율 변화를 기반으로 하며 실시간 반응성이 높은 것이 장점이다.

- PHT

PHT(Page-Hinkley Test)는 평균의 누적 변화량을 추적하여 갑작스러운 평균의 변화를 감지하는 전통적인 통계 기반 기법이다[10][19]. 특히 데이터의 평균 변화에 민감하게 반응하며 실시간 스트리밍 데이터에 적용하기 용이하다. 이 방법은 누적된 차이값이 특정 임계값을 넘는 경우 드리프트가 발생한 것으로 판단한다.

- ADWIN

ADWIN(ADaptive WINdowing)은 가변 크기의 인스턴스 슬라이딩 윈도우(W)를 유지하면서 스트림에서 실시간으로 분포 변화를 감지하는 기법이다[20]. 데이터가 들어올 때마다 현재 윈도우를 가능한 한 오래 유지하되 윈도우를 둘로 나뉘었을 때 양쪽 부분의 평균이 통계적으로 다르면 오래된 윈도우 부분을 제거한다. 동적으로 조정된 두 개의 하위 윈도우가 저장되어 기존 데이터와 최신 데이터를 나타내고 이들 윈도우의 평균 차이가 임계값 보다 높을 때 드리프트가 발생했다고 판단한다. 이 기법은 시간 경과에 따른 데이터의 점진적인 변화를 탐지하는 데 유효하다.

2.2.3 AutoML 기반 탐지 기법

AutoML 기반 드리프트 탐지 및 적응 기법 연구는 최근 활발히 진행되고 있으나 IoT환경에서의 드리프트나 산업용 센서 드리프트 탐지 등 특정 분야에 적용한 사례가 대부분이다. Celik는 AutoML 환경에서 베이지안 최적화, 유전자 프로그래밍 등을 사용하여 개념 드리프트에 대응하기 위한 detect & retrain, detect & warm-start 등 다양한 적응 전략들을 제안하고 실험적으로 평가하였다[21]. 이러한 연구들은 제한된 범위 내에서 드리프트 탐지나 드리프트의 대응 방법에 초점을 맞추고 있다.

3. D²MR: Drift Detection Method Recommender

본 장에서는 기존 연구 자료를 체계적으로 분석하여 데이터 특성과 적용 가능한 드리프트 탐지 기법을 분류하고 분류 기준을 제시한다. 이러한 데이터 특성과 발생 가능한 드리프트를 매핑하여 적합한 드리프트 탐지 기법을 추천하는 D²MR(Drift Detection Method Recommender) 프레임워크를 제안한다.

3.1 데이터 특성 기반 드리프트 탐지 기법

ML 모델의 성능은 데이터의 품질과 특성에 큰 영향을 받는다. 드리프트를 보다 정확하게 탐지하기 위해서는 데이터 특성에 따라 적합한 탐지 기법을 선정하는 것이 중요하다. 이를 위해 본 연구에서는 데이터 특성을 대/소분류로 구분하고 각 분류를 정량적 또는 정성적 기준을 제시한다. 본 연구에서는 분류한 데이터 특성을 기술하고 표 1과 같이 데이터 특성별 적용 가능한 드리프트 탐지 기법을 분류한다.

3.1.1 Label Availability

데이터의 라벨 존재(≥ 90%) 여부에 따라 탐지 기법이 달라진다. 라벨이 있는(labeled) 경우는 분류 오류(classification error)나 혼동행렬(confusion matrix) 등 지도 학습 기반 지표를 활용한다. 라벨이 없는(unlabeled) 경우는 비지도 학습 기반으로 입력 데이터의 분포 변화 또는 임베딩 기반 시각적 변화를 이용하여 드리프트를 탐지한다. MMD, LPIPS, FID 등의 기법이 이에 해당한다.

3.1.2 Data Type

데이터 유형은 정형과 비정형 여부에 따라 통계기법과 딥러닝 기반의 탐지 기법을 구분하여

표 1. 데이터 특성 기반 드리프트 탐지 기법
Table 1. Drift Detection Method based on Data Characteristics

High-level Category	Sub-level Category	Criteria	Applicable Drift Detection Methods
Label Availability	Labeled	Supervised (labels available)	PSI, KL-Divergence, ADWIN, DDM, EDDM, Page-Hinkley, HDDDM, Classifier-based
	Unlabeled	No label information	MMD, LPIPS, ADWIN
Data Type	Structured	Tabular (Numeric or categorical tables)	PSI, KL-Divergence, JS-Divergence, KS-Test, ADWIN, DDM, EDDM, Page-Hinkley
	Unstructured	Image, text, audio, embeddings	FID, LPIPS, MMD
Dataset Size	Large Dataset	$\geq 10,000$ samples	PSI
	Small Dataset	$\leq 1,000$ samples	KS-Test
	Fast Stream (Online/Streaming)	Continuous, sequential data	ADWIN, DDM, EDDM, Page-Hinkley
Dimensionality	Low Dimensionality	< 20 features	KS-Test, KL-Divergence
	High Dimensionality	> 100 features / embeddings	JS-Divergence, FID, MMD
Distance Computability	Distance Computable	Euclidean, cosine, kernel, etc.	KL-Divergence, JS-Divergence, KS-Test, FID, MMD
	Distance Not Directly Computable	Perceptual similarity only	LPIPS
Distribution Type	Normal Distribution	Gaussian assumption	KL-Divergence
	Non-Normal Distribution	No distribution assumption	JS-Divergence, MMD

적용한다. 정형 데이터는 범주형이나 수치형과 같이 정의가 명확한 컬럼이나 속성이 포함된 구조화된 테이블 형태의 데이터이다. 정형 데이터인 경우는 분포 변화를 평가하기 위해 대부분 통계기반의 드리프트 탐지 기법이 사용된다. 이미지나 오디오, 텍스트, 자연어와 같은 비정형 데이터는 주로 딥러닝을 통해 임베딩을 추출하는 FID, LPIPS 등 탐지 기법을 적용한다.

3.1.3 Dataset Size

데이터셋 크기는 수집된 데이터의 샘플 수이다. 데이터셋 크기는 탐지 기법에 따라 통계적 유의성과 신뢰도가 달라질 수 있다. 일부 드리프트 탐지 기법은 충분한 통계량이 확보되어야 유효하고 작은 크기의 데이터셋은 오류률이 증가하

거나 드리프트 탐지가 어렵다. 본 연구에서는 기존 연구들을 참고와 경험적인 실험을 기반으로 샘플 수 10,000을 기준으로 소규모 데이터셋과 대규모 데이터셋으로 구분한다. 대용량 데이터셋은 PSI 기법이 유효하고 소규모 데이터셋은 비모수(non-parametric) 접근법을 사용하는 KS-Test, MMD 등을 적용하는 것이 적절하다.

스트리밍 환경에서 실시간으로 입력되는 데이터는 평균의 변화나 오류율 변화를 빠르게 감지해야 한다. 이런 경우는 데이터 도착률 조건은 초당 천 개 이상의 레코드가 되어야 유효하다. ADWIN, DDM 등 탐지 기법은 이러한 조건에 맞게 sliding window 또는 누적 통계 기반으로 동작한다.

3.1.4 Dimensionality

데이터 차원(Dimensionality)은 데이터셋을 구성하는 특성(feature)의 개수이다. 고차원은 특성이 100개 이상인 경우 저차원은 특성이 20개 이하인 데이터셋으로 구분한다. 고차원 데이터셋은 임베딩 기반 기법이나 차수를 축소한 후 탐지를 수행하는 탐지 기법을 사용한다. 저차원의 데이터셋은 전통적인 통계 기반 기법을 적용한다.

3.1.5 Distance Computability

데이터의 통계적 분포 차이를 측정 가능성 여부에 따라 적용하는 탐지 기법이 달라질 수 있다. 수치형 데이터나 벡터 데이터인 경우 유클리디안 거리나 정규화된 임베딩 거리 계산이 가능하기 때문에 대부분 통계 기반 탐지 기법이 사용 가능하다. 이미지나 오디오 데이터와 같은 개념적 유사성이 거리로 직접 측정되지 않는 경우 사전 학습된 신경망의 deep feature 기반 거리를 활용하는 LPIPS 기법을 사용한다.

3.1.6 Distribution Type

데이터 분포의 정규성(distribution normality)은 데이터의 가우시안 분포 여부에 따라 탐지 기법을 적용한다. 데이터가 정규 분포인 경우는 확률분포 간 차이를 측정하는 KL-Divergence 기법을 적용한다. 데이터 비정규 분포인 경우는 데이터가 정규성이 없이도 분포 간 차이를 비교할 수 있는 Wasserstein distance, MMD 등 비모수 탐지 기법을 적용한다.

3.2 데이터 특성 기반 드리프트 유형

3.2.1 Drift Type Classification

본 연구에서 드리프트 유형은 데이터 드리프트, 개념 드리프트, 사전확률 드리프트로 분류한다. 데이터 드리프트는 $P(x)$ 가 변하는 경우로

ML 모델과 무관하게 데이터 자체의 분포를 분석한다. 개념 드리프트는 입/출력 관계($P(y | x)$)가 변하는 경우로 데이터는 같아도 예측 결과가 달라지는 것을 평가한다. 사전확률 드리프트는 클래스 분포 $P(y)$ 가 변하는 경우이다. D²MR 프레임워크는 단계별로 드리프트 탐지 기법의 차원 축소 접근법(reduction approach)을 적용하여 최종 드리프트 탐지 기법이 추천된다. 따라서 드리프트 유형을 추론하기 위해 데이터 특성 기반으로 드리프트 유형을 분류한다. 예를 들어 라벨이 있는 데이터의 경우 $P(x)$, $P(y)$, $P(y|x)$ 변화를 모두 추적 가능하여 모든 발생 가능한 드리프트 유형을 알 수 있다. 반면, 라벨이 없는 데이터의 경우 $P(y|x)$ 는 직접적으로 추정하기 어렵다. 따라서 $P(x)$ 기반의 변화를 우선 감지하여 발생 가능한 탐지 기법을 추론한다. 이와 같이 각 데이터 특성을 분석하여 추적/감지해야 하는 드리프트 유형을 표 2와 같이 분류한다.

표 2. 데이터 특성 기반 드리프트 유형
Table 2. Drift Type based on Data Characteristics

High-level Category	Sub-level Category	Drift Types to Detect
Label Availability	Labeled	Data Drift Concept Drift Prior Probability Drift
	Unlabeled	Data Drift
Data Type	Structured	Data Drift Prior Probability Drift
	Unstructured	Change in $P(y x)$ Sudden Drift Gradual Drift Incremental Drift Prior Probability Drift
Dataset Size	Large Dataset	Data Drift Label Shift($P(y)$)
	Small Dataset	Data Drift Prior Probability Drift
	Fast Stream (Online/Streaming)	Sudden Drift Gradual Drift Incremental Drift
Dimensionality	Low	Data Drift

	Dimensionality	
	High Dimensionality	Data Drift Prior Probability Drift
Distance Computability	Distance Computable	Data Drift Prior Probability Drift
	Distance Not Directly Computable	Data Drift
Distribution Type	Normal Distribution	Data Drift Prior Probability Drift
	Non-Normal Distribution	Data Drift Prior Probability Drift

3.2.2 Drift Detection Method by Drift Type

각 드리프트 유형별 드리프트 탐지 기법을 분류한다. 데이터 드리프트 탐지 기법에는 통계적 분포 비교, 거리 기반 및 윈도우 기반 탐지 기법이 있다. 개념 드리프트 탐지 기법에는 오류율 기반 및 통계적 누적 검정 기반 탐지 기법이 있다. 사전확률 드리프트 기법은 클래스 분포를 평가하거나 라벨 기반 윈도우 탐지기법이 있다. 드리프트 유형별 대표적인 드리프트 탐지 기법을 표 3과 같이 분류한다.

표 3. 드리프트 유형 별 드리프트 탐지 기법
Table 3. Drift Detection Method by Drift Type

Drift Type	Criteria	Applicable Drift Detection Methods
Data Drift	Covariate shift (P(x))	PSI, KS-Test, KL/JS-Divergence, Wasserstein Distance, MMD, ADWIN
Concept Drift	Change in P(y x)	ADWIN, DDM, EDDM, HDDM, KSWIN
	Sudden Drift	Page-Hinkley Test, CUSUM, ADWIN
	Gradual Drift	FDDM, EDDM, ADWIN
	Incremental Drift	ADWIN, HDDM, EWMA
Prior Probability Drift	Label Shift (P(y))	Chi-Square Test, KL/JS-Divergence, HDDM, STEPD

3.3 D²MR Framework

D²MR 프레임워크는 운영되고 있는 ML 시스템에 입력되는 데이터의 특성을 자동 추출하고 감지해야 할 드리프트 유형을 추론하여 이를 기반으로 가장 적합한 드리프트 탐지 기법을 추천한다. D²MR 프레임워크는 (1) Automatic Feature Extraction, (2) Drift Type Extraction, (3) Filtering Candidate Detector, (4) Drift Recommender, (5) Drift Detection Executor의 다섯 개의 요소로 구성된다.

3.3.1 Automatic Feature Extraction

자동 특성 추출(AFE: Automatic Feature Extraction) 구성 요소는 입력된 데이터셋에서 드리프트 탐지에 요구되는 특성을 자동으로 추출한다. 데이터 특성은 3.1절에서 정의한 기준을 기반으로 다음의 정보를 추출한다.

- 라벨 유무: supervised/unsupervised
- 데이터 유형: 정형/비정형, 수치형/범주형 등
- 데이터 크기: 샘플 수의 양
- 데이터의 차원: 데이터 특성수, 고차원 여부
- 거리 계산가능성: 유사도 계산 가능 여부
- 데이터 분포: 정규 분포 유무

이러한 특성은 드리프트 탐지 기법의 필터링 및 추천의 기준으로 사용된다.

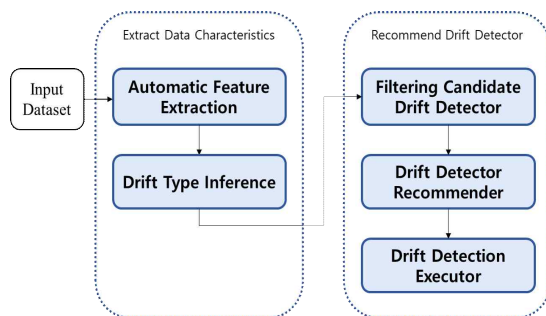


그림 2. D²MR 프레임워크
Fig. 2. D²MR Framework

3.3.2 Drift Type Inference

드리프트 유형 추론(DTI: Drift Type Inference)은 표 2를 기반으로 추출된 데이터 특성에 따라 해당 데이터에 따라 탐지해야 하는 드리프트 유형을 추론한다.

- 데이터 드리프트: 입력 변수의 분포 변화
 - 개념 드리프트: 입력-출력 간 관계의 변화
 - 급격한/점진적/누적/반복적 드리프트: 드리프트 발생 시간적 패턴 기반
 - 사전확률 드리프트: 출력의 분포 변화
- 추론 결과는 탐지 기법 추천을 체계적으로 추천할 수 있는 중요한 정보가 된다.

3.3.3 Filtering Candidate Detector

후보 탐지기 필터기(FCD: Filtering Candidate Detector)는 추출한 입력 데이터의 특성과 추론한 드리프트 유형을 기반으로 관련된 탐지 기법들을 선정한다. FCD는 다수의 드리프트 탐지 기법에서 적합한 탐지기를 선정하는 축소 접근법의 단계로 최종적으로 추천할 드리프트 탐지 기법의 후보군을 선정하는 단계이다. 본 연구에서는 표 1, 3에서 분류한 드리프트 탐지 기법을 데이터베이스로 구축하여 이를 참조하여 규칙 기반 기법을 사용하여 후보군을 선정한다. 예를 들어, 데이터가 수치형이고 라벨이 없다면 KS-Test, PSI, MMD기법 등이 최종 드리프트 탐지기의 후보군이 된다.

3.3.4 Drift Detector Recommender

드리프트 탐지기 추천(DDR, Drift Detection Recommender)는 FCD에서 필터링된 후보 탐지 기법들의 성능을 평가하여 그 중에서 성능이 가장 좋은 드리프트 탐지 기법을 선정하여 최종적으로 추천한다. 본 연구에서는 성능 지표를 AUC, Accuracy, F1 score, Precision, Recall를 사용하였다. 그러나 Detection Delay, False

Alarm Rate 등의 지표를 추가 사용하거나 데이터 특성에 따라 지표의 가중치를 달리하여 평가할 수 있다.

3.3.5 Drift Detection Executor

드리프트 탐지 실행기(DDE, Drift Detection Executor)는 DDR에서 추천받은 드리프트 탐지 기법을 실행한다. DDE는 ML 시스템을 모니터링 하면서 드리프트가 탐지 여부를 ML 시스템에 결과를 제공할 수 있는 ML 생명주기와의 통합 가능성을 제시한다.

표 4. D²MR 슈도코드
Table 4. D²MR Pseudocode

Algorithm: D ² MR Framework
<p>Input:</p> <ul style="list-style-type: none"> input_data: operational dataset (current or incoming) MD: set of meta data {MD.label, ..., MD.normalization} DDM_table: metadata of all drift detection methods and their compatibility <p>Output:</p> <ul style="list-style-type: none"> R_detectors: recommended drift detectors D_mode: the result of drift detection
<p>Step 1: Automatic Feature Extraction</p> <pre> Features₁ ← has_labels(input_data, MD.label) Features₂ ← extract_data_type(input_data, MD.data_type) Features₃ ← count_samples(input_data, MD.data_size) Features₄ ← estimate_dimension(input_data, MD.dimensionality) Features₅ ← can_define_distance(input_data, MD.distance) Features₆ ← detect_normalization(input_data, MD.normalization) </pre>
<p>Step 2: Drift Type Inference</p> <pre> SelectedDriftType ← get_drift_type(Features) collect_ddm1 ← ∅ for each method m in DDM_table do if SelectedDriftType ∈ m.compatible_drift_types collect_ddm1 ← collect_ddm1 ∪ {m} end if end for </pre>
<p>Step 3: Filtering Candidate Drift detector</p> <pre> collect_ddm1 ← ∅ for each method m in DDM_table do if Feature ∈ m.compatible_drift_types </pre>

```

        collect_ddm1 ← collect_ddm1 ∪ {m}
    end if
end for
collect_ddm2 ← ∅
for each method m in DDM_table do
    if SelectedDriftType ∈ m.compatible_drift_types
        collect_ddm2 ← collect_ddm2 ∪ {m}
    end if
end for
candidate_detectors ← filtering_detectors(collect_ddm1,
collect_ddm2)

```

Step 4: Drift Detector Recommender
 eval_performs ← ∅
 for each method m in candidate_detectors do
 m.perform ← evaluate_performance(input_data, m)
 eval_performs ← eval_performs ∪ {m.perform}
 end for
 R_detector ← best_performance(eval_performs)

Step 5: Drift Detector Executor
 D_mode ← APPLY_DETECTOR(input_data, R_detector)

Return R_detector, D_mode

4. Case Study

본 장에서는 D2MR 프레임워크의 유효성을 검증하기 위해 정형 데이터와 이미지 데이터셋을 활용하여 실험을 진행한다.

4.1 실험 설정

4.1.1 CIFAR-10 Dataset

CIFAR-10은 10개 클래스에 동일한 개수의 이미지가 포함된 데이터셋으로 구성되어 있다[22]. CIFAR-10은 32x32 픽셀크기의 약 60,000개의 RGB 이미지로 구성되어 있으며 클래스 간 분포가 균일하고 시각적 스타일 변화가 주요한 드리프트 요인이 된다.

4.1.2 UCI Wine Quality Dataset

UCI Wine Quality 데이터는 다양한 화학적 특성과 그에 따른 품질 등급에 대한 정보가 포함되어 있는 수치형 특성으로 구조화된 데이터셋이다

[23]. UCI Wine 데이터는 저차원의 정규분포 특성이 강하게 나타나는 특성이 있다. 모든 유형의 드리프트 유형의 가능성이 추론되었다.

4.2 실험 결과

4.2.1 데이터 특성 추출

CIFAR-10와 UCI Wine 데이터셋에서 추출한 데이터 특성은 표 5와 같다.

표 5. 데이터 특성: CIFAR-10 vs UCI Wine
 Table 5. Data Characteristics: CIFAR-10 UCI Wine

Data Characteristics	CIFAR-10	UCI Wine
Label Available	labeled	labeled
Data Type	unstructured /image	structured /tabular
Dataset Size	large	small
Dimensionality	high	low
Distance Computability	computable	computable
Distribution Type	non-normal	normal
Inferred Drift Type	data/prior probability drift	data/concept/prior probability drift

4.2.2 후보 드리프트 탐지 기법

1) CIFAR-10 Dataset

CIFAR-10 데이터셋은 클래스간의 분포가 균일하고 시각적 변화가 주요한 드리프트 요인이다. 데이터의 픽셀 값이 0~255 범위의 이산분포를 따르고 각 이미지가 동일한 차원과 구조를 가지므로 유클리드 거리나 코사인 유사도 등을 계산할 수 있다. 픽셀 기반의 이미지 벡터 데이터가 이미지 간의 유사성을 제대로 반영하지 못하는 경우 이를 보완하기 사전 학습된 딥러닝 기반 지표를 활용할 수 있다. 이러한 데이터 특성들을 기반으로 드리프트 유형을 추론하여 픽셀 기반 분포 변화에 민감한 FID, MMD, LPIPS 기법이

추천 후보군으로 필터링되었다.

2) UCI Win Quality Dataset

UCI Wine Quality은 약 4,898개의 샘플과 11개의 수치형 독립 변수로 구성된 저차원의 데이터셋이다. 와인 데이터셋의 대부분 피처는 정규 분포를 따르지 않고 있고 와인 품질 점수(0~10)가 타겟변수가 되어 회귀 또는 다중 클래스 분류 문제로 라벨링이 되어 있다. 이러한 데이터 특성을 기반으로 통계 기반 분포 비교에 적합한 PSI, KS-Test 등이 후보군으로 필터링되었다.

4.2.3 추천 드리프트 탐지 기법

데이터셋에 적용할 수 있는 후보 드리프트 탐지 기법들의 성능을 평가하여 성능이 가장 좋은 탐지 기법을 최종적으로 추천한다.

1) CIFAR-10 Dataset

AUC와 F1-Score 평가지표를 사용하여 각 후보 기법들의 데이터 분포 변화 감지 능력을 평가하였다. CIFAR-10 테스트셋에 다양한 색상의 변화를 적용하여 드리프트를 시뮬레이션하였다. 그림 3은 후보 드리프트 탐지 기법들의 성능 평가한 결과이다. LPIPS기법도 우수한 성능을 보였으나 FID 기법이 AUC/F1-Score 모두 높은 성능으로 최종 탐지 기법으로 추천되었다.

LPIPS와 FID 기법 모두 성능 차이가 크지 않으면서 우수한 성능을 보이므로 CIFAR-10 테스트셋과 유사한 데이터 특징을 갖는 실제 데이터 환경에서는 두 기법을 병행하여 모니터링하는 전략도 활용할 수 있다.

2) UCI Wine Quality Dataset

UCI Wine 데이터셋에 추천된 후보 기법에 대해 정확도를 분석한 결과 KS-Test가 높은 성능을 보였다. KS-Test는 드리프트된 특징점 탐지에서 뛰어난 성능을 보였으며 실험한 5개의 드리

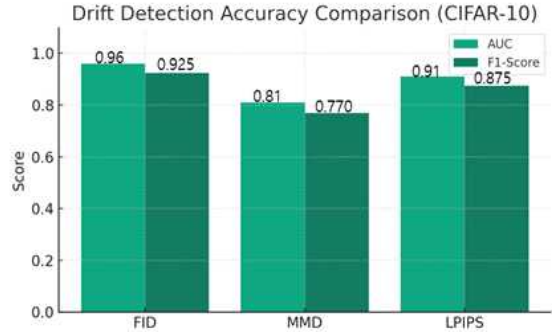


그림 3. 후보 드리프트 탐지기법 정확도(CIFAR-10)
Fig. 3. Candidate Drift Detector Accuracy(CIFAR-10)

프트된 특징점 중 4개를 정확하게 식별했다. 반면, PSI는 표준 임계값($PSI > 0.1$) 미만의 드리프트된 특징을 하나도 탐지하지 못했다. 따라서 KS-Test가 UCI Wine 데이터셋에서 드리프트를 탐지하는 데 더 효과적이고 신뢰할 수 있는 것으로 나타났다.

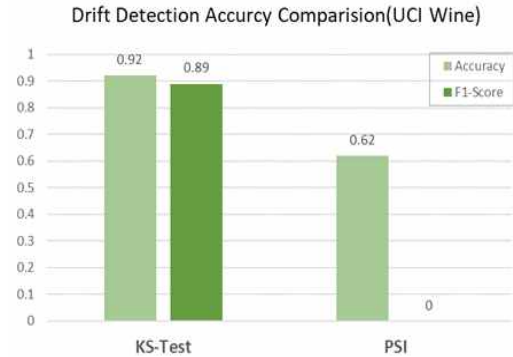


그림 4. 후보 드리프트 탐지기법 정확도(UCI Wine)
Fig. 4. Candidate Drift Detector Accuracy(UCI Wine)

4.3 토의

D²MR 프레임워크는 그림 5와 같이 ML 생명주기와 통합 확장 가능하다. D²MR은 운영하고 있는 ML 시스템의 전처리 이후의 데이터셋을 입력받아 데이터 특성을 분석하고 적합한 드리프트 탐지 기법을 추천한다. 이는 ML 시스템을 모니

터링하면서 추천받은 탐지 기법을 사용하여 드리프트 탐지 여부를 분석하고 드리프트가 감지되면 ML 모델을 재학습할 수 있도록 ML 시스템과 통합하여 성능을 지속적으로 유지할 수 있다.

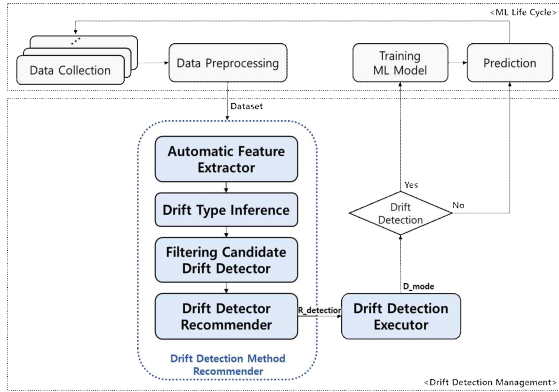


그림 5. D²MR 과 ML 생명주기
Fig. 5. D²MR and ML Lifecycle

5. 결론

본 연구에서는 머신러닝 시스템의 드리프트 현상에 대한 데이터 특성을 고려하지 않은 드리프트 탐지 기법의 선정 및 드리프트 탐지 기법을 선정하는 가이드라인의 부재를 해결하기 위한 드리프트 탐지 기법 추천 방법론을 제시하였다. 본 연구는 드리프트에 대응하기 위해 데이터 특성에 기반한 드리프트 탐지 기법을 추천하는 프레임워크 D²MR을 제안하였다. 제안한 D²MR 프레임워크는 다음과 같은 특징을 가진다.

첫 번째, 데이터 특성의 체계적인 분류와 기준으로 제시하였다. 이를 기반으로 데이터 특성 자동 추출과 추론한 드리프트 유형을 기반으로 탐지 기법 후보군을 체계적으로 축소함으로써 최종적으로 드리프트 탐지 기법을 추천한다. 두 번째, 다양한 드리프트 탐지 기법을 데이터 특성과 드리프트 유형 별로 분류하여 탐지 기법의 적합

성을 분석하고 매핑 테이블을 구성하였다. 세 번째, 실험을 통해 CIFAR-10 (이미지데이터)과 UCI Wine Quality (정형데이터)과 같은 서로 다른 유형의 데이터셋에 대해 적합한 탐지 기법이 추천됨을 보여주었다.

본 연구에서는 데이터 특성 기반으로 드리프트 탐지 기법 추천을 체계적으로 접근할 수 있는 방법론을 제안하였다. 그러나 실험 데이터가 표준 벤치마크 데이터셋을 사용하여 드리프트를 발생시켜 제안한 방법론을 검증하였기 때문에 몇 가지 한계와 추가적인 연구가 요구된다.

첫 번째, 검증 데이터를 실제 현실에서 운영하는 데이터셋을 사용하여 보다 더 세분화된 데이터 요소 도출과 분류 기준의 임계값 설정이 요구된다. 두 번째, D²MR 프레임워크에서는 하나의 드리프트 탐지 기법을 추천하고 있다. 그러나 실제 운영 환경에서는 다양한 드리프트 유형이 발생하는 만큼 복수의 탐지 기법을 추천하고 병행 실행하면서 실시간으로 성능을 평가하는 통합 기능이 요구된다. 세 번째, 실시간 데이터 환경에서 ML 생명주기와 통합하여 AutoML 기반의 동적 드리프트 탐지 기법 추천 시스템에 대한 연구가 요구된다.

이 논문은 정부(과학기술정보통신부)의
재원으로
정보통신기획평가원-지역지능화혁신인재양성
사업의 지원을 받아 수행된 연구임
(IITP-2025-RS-2022-00156334)

참고 문헌

[1] C. Zhang et al., "Understanding deep learning requires rethinking generalization", ICLR 2017, DOI: 10.48550/arXiv.1611.03530

- [2] Firas Bayram et al., “From Concept Drift to Model Degradation: An Overview on Performance-Aware Drift Detectors”, *Knowledge-Based Systems*, vol. 245, Jun. 2022, DOI: 10.1016/j.knosys.2022.108632
- [3] A. Tsymbal, “The Problem of Concept Drift: Definitions and Related Work”, Computer Science Department, Trinity College, Dublin, 106(58), 2004, https://www.researchgate.net/publication/228723141_The_Problem_of_Concept_Drift_Definitions_and_Related_Work
- [4] R.P.J.C. Bose et al., “Handling concept drift in process mining”, *CAiSE 2011, LNCS 6741*, pp. 391-405, 2011, DOI: 10.1007/978-3-642-21640-4_30
- [5] Tri Minh Triet Pham et al., “Time to Retrain? Detecting Concept Drifts in Machine Learning Systems”, *arXiv*, 2024, DOI: 10.48550/arXiv.2410.09190
- [6] Jie Lu et al., “Learning under Concept Drift: A Review”, *IEEE Transactions on Knowledge and Data Engineering*, 31(12), pp.2346-2363, Dec. 2019, DOI: 10.1109/TKDE.2018.2876857
- [7] Moreno-Torres et al., “A unifying view on dataset shift in classification”, *Pattern Recognition*, 45(1), pp.521-530, Jan 2012, DOI: 10.1016/j.patcog.2011.06.019
- [8] B. Yurdakul and J. Naranjo, “Statistical Properties of Population Stability Index”, *Journal of Risk Model Validation*, 14(4), pp.89-100, 2020. DOI: 10.21314/JRMV.2020.227
- [9] S. Kullback and R. A. Leibler, “On information and sufficiency”, *Institute of Mathematical Statistics*, 22(1), pp.79-86, 1951. <https://www.jstor.org/stable/2236703>
- [10] F. Nielsen, “On a Generalization of the Jensen-Shannon Divergence and the Jensen-Shannon Centroid”, *MDPI Entropy*, 21(5), 2019. DOI: 10.3390/e21050485
- [11] J. Gama et al., “A Survey on Concept Drift Adaptation,” *ACM Computing Surveys*, 46(4), Article:44, pp.1-37, Mar. 2014. DOI: 10.1145/2523813
- [12] A. Gretton, et al., “A Kernel Two-Sample Test”. *Journal of Machine Learning Research*, vol. 13, pp.723-773, 2012, DOI: 10.5555/2188385.2188410
- [13] Min Jin Chong and David Forsyt “Effectively Unbiased FID and Inception Score and where to find them”, *CVPR 2020*, DOI: 10.48550/arXiv.1911.07023
- [14] M. Heusel et al., “GANs trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium”, *NeurIPS*, 2017, <https://neurips.cc/Conferences/2017>
- [15] R. Zhang et al. “The Unreasonable Effectiveness of Deep Features as a Perceptual Metric” *IEEE/CVF Computer Vision and Pattern Recognition*, 2018, DOI: 10.1109/CVPR.2018.00068
- [16] J. Gama et al., “Learning with Drift Detection”, *Lecture Notes in Computer Science (LNAI 3171)*, pp.286-295, 2004, DOI: 10.1007/978-3-540-28645-5_29
- [17] M. Baena-García et al., “Early Drift Detection Method”, In *Proceedings of the International Workshop on Knowledge Discovery from Data Streams*, vol. 6, pp. 77-86, 2006
- [18] A. Pesaranghader et al. “Reservoir of Diverse Adaptive Learners and Stacking Fast Hoeffding Drift Detection Methods for Evolving Data Streams”. *Machine Learning*, vol. 107, pp.1711-1743, 2018, DOI: 10.1007/s10994-018-5719-z
- [19] E.S. Page, “Continuous Inspection Schemes”. *Biometrika*, 41(1/2), pp.100-115, 1954, DOI: 10.2307/2333009
- [20] Albert Bifet and Ricard Gavaldà, “Learning from Time-Changing Data with Adaptive Windowing”, *Proceedings of the Seventh SIAM International Conference on Data Mining*, pp. 443-448. 2007, DOI: 10.1137/1.9781611972771.42
- [21] B. Celik and J. Vanschoren, “Adaptation

Strategies for Automated Machine Learning on Evolving Data”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(9), pp. 3067-3078, 2021, DOI: 10.1109/TPAMI.2021.3062900

- [22] CIFAR-10 Dataset, [Internet],
<https://www.cs.toronto.edu/~kriz/cifar.html>
- [23] UCI Wine Quality Dataset, [Internet],
<https://archive.ics.uci.edu/dataset/186/wine+quality>

저 자 소 개



최옥주(Okjoo Choi)

2008.02 숙명여자대학교 컴퓨터학과 박사
1990.8-1996.3 LG생산기술원 주임연구원
1996.7-2009.8 한국오라클 수석컨설턴트
2009.9-2022.12 카이스트 전산학부 연구교수
2023.10-2024.2 강원대학교 산학협력중점교수
2024.3-현재 배재대학교 AI-SW공학부 조교수
<주관심분야> 빅데이터 분석, 데이터품질, 소프트웨어품질, 프로젝트 관리, AI안정성