논문 2024-1-7 http://dx.doi.org/10.29056/jsav.2024.03.07

머신러닝 드리프트에 대한 2-단계 데이터 품질 평가 방법론

최옥주*, 김유경**†

Two-steps Data Quality Assessment Methodology for Handling Drift of Machine Learning

Okjoo Choi*, Yukyong Kim**†

요 약

빅 데이터 분석이나 머신러닝 모델과 같은 데이터 기반의 정보 기술 분야에서 데이터 품질은 시스템 전체의 품질과 직접적으로 연결된다. 특히 머신러닝 모델의 훈련에 사용된 데이터의 속성은 시간이 지나면서 변화하게 되는데, 이로 인해 모델의 정확도가 떨어지거나 설계된 것과 다르게 작동할 수 있게 된다. 이러한 현상을 드리프트(drift)라고 한다. 드리프트는 데이터 수집 문제나 시장의 변동성 등 다양한 이유로 인해 발생할 수 있다. 데이터 드리프트는 즉시 감지되기 어렵고, 예측이 부정확해 지기 때문에 예측을 기반으로 내린 비즈니스 결정에 어려움을 겪을 수 있다. 드리프트를 관리하기 위해 필요한 작업은 드리프트의 유형이나 범위 및 성격에 따라 달라진다. 적절한 조치를 취하려면 드리프트 식별 뿐만 아니라 데이터 품질 관리 및 평가와 함께 드리프트 비율에 대한 임계값 설정 및 사전 경고 구성을 위한 반복 가능한 절차를 확립하는 것이 중요하다. 본 논문에서는 머신러닝 프로젝트에서 발생하는 드리프트 문제를 데이터의 품질평가 메트릭을 통해 관리할 수 있는 2단계 데이터 품질평가 프레임워크를 제안하고, 드리프트 탐지를 위한 드리프트 유형에 따른 평가 메트릭스와 평가 절차를 정의한다.

Abstract

Data quality of data-based information technologies such as big data analysis and machine learning directly affects the quality of the entire system. In particular, the properties of the data used to train machine learning models change over time, causing the model to become less accurate or behave differently than it was designed to. This phenomenon is called drift. Drift can occur for a variety of reasons, including data collection issues or market volatility. Data drift is difficult to detect immediately and can lead to inaccurate predictions, compromising business decisions based on it. The actions required to manage drift will depend on the type, extent, and nature of the drift. To take appropriate action, it is important to establish repeatable procedures for identifying drift, controlling and assessing data quality, setting thresholds for drift rates, and configuring proactive warnings. In this paper, we propose a two-step data quality assessment framework that can manage drift problems that occur in machine learning projects through data quality assessment indicators. In addition, evaluation indices and evaluation procedures according to drift type for drift detection are also defined.

한글키워드 : 데이터 품질 평가, 데이터 품질 메트릭, 데이터 드리프트, 개념 드리프트

keywords : Data quality assessment, Data quality metric, Data Drift, Concept Drift

* 배재대학교 AI·소프트웨어공학부
** 숙명여자대학교 기초공학부
† 교신저자: 김유경(email: ykim.be@sookmyung.ac.kr)

접수일자: 2024.03.05. 심사완료: 2024.03.16.
게재확정: 2024.03.20.

www.kci.go.kr

1. 서 론

데이터 품질은 모든 머신러닝 프로젝트의 성공 요인 중 가장 중요한 요소라고 할 수 있다. 데이터 품질이 좋지 않으면, 모델의 부정확성 문제뿐만 아니라 예측 결과에 대해 신뢰할 수 없게 되는 문제가 발생한다. 이와 함께 머신러닝 모델의 훈련에 사용된 데이터의 속성이 시간이 지나면서 변화하게 되는데, 이로 인해 모델의 정확도가 떨어지거나 설계된 것과 다르게 작동할 수 있게 된다. 이러한 현상을 드리프트(drift)라고 한다. 즉, 머신러닝 모델이 사용되는 환경의 변화로 인해 모델의 정확한 예측 능력이 저하되는 것이다[1].

데이터 드리프트 문제는 시간 경과에 따른 머신러닝 모델의 정확도 변화 문제로, 일반적으로 모델이 훈련에 사용되는 데이터와 상당히 다른 데이터에 대해 추론을 실행하기 때문에 발생한다. 예를 들어, 데이터를 기반으로 회사의 주가를 예측하도록 설계된 머신러닝 모델이 안정적인 시장의 데이터로 훈련되었다면, 처음에는 좋은 결과를 얻을 수 있지만, 시간이 지나면서 시장의 변동성이 커지게 된다면 데이터의 통계적 속성이 변하기 때문에 더 이상 주가를 정확하게 예측하지 못할 수 있다.

드리프트는 데이터 수집 문제나 시장의 변동성 등 다양한 이유로 인해 발생할 수 있다. 데이터 드리프트는 즉시 감지되기 어렵고, 예측이 부정확해 지기 때문에 예측을 기반으로 내린 비즈니스 결정에 어려움을 겪을 수 있다. 드리프트를 관리하기 위해 필요한 작업은 드리프트의 유형이나 범위 및 성격에 따라 달라진다. 어떤 상황에서는 새 데이터를 사용하여 모델을 다시 학습하여 드리프트를 관리할 수 있지만, 다른 경우에는 처음부터 다시 시작해야 할 수도 있다. 머신러닝 모델의 예측이 더 이상 도움이 되지 않을 때 드리프트 문제가 발생하는 것은 단순히 데이터 뿐만 아니라 모델 자체의 문제일 수도 있다. 모델은 변화하는 상황을 인식하지 못하기 때문이다. 적절한 조치를 취하려면 드리프트 식별 뿐만 아니라 데이터 품질 관리 및 평가와 함께 드리프트 비율에 대한 임계값 설정 및 사전 경고 구성을 위한 반복 가능한 절차를 확립하는 것이 중요하다.

따라서 본 논문에서는 머신러닝 프로젝트에서 발생하는 드리프트 문제를 데이터의 품질평가 메트릭을 통해 해결해 보고자 한다. 이를 위해 2단계 데이터 품질평가 프레임워크를 제안하고, 드리프트 탐지를 위한 평가 메트릭스와 평가 절차를 정의한다.

2. 관련 연구

2.1 드리프트 개념과 유형

드리프트는 크게 개념 드리프트(Concept drift)와 데이터 드리프트(Data drift)로 나뉜다. 먼저 모델 드리프트(Model drift)라고도 부르는 개념 드리프트는 머신러닝 모델이 수행하도록 설계된 작업이 시간이 지남에 따라 변경될 때 발생한다. 예를 들어, 이메일 내용을 기반으로 스팸을 탐지하도록 머신러닝 모델을 훈련했다고 하면, 사람들이 받는 스팸메일의 유형이 크게 변경되면 머신러닝 모델은 더 이상 스팸을 정확하게 감지할 수 있다. 개념 드리프트는 다음 그림 1과 같이 크게 4가지 범주로 나눌 수 있다[1].

모델의 성능이 저하되는지 확인하기 위해서 정확도, 오류율, 또는 클릭율과 같은 KPI(Key Performance Indicator)메트릭을 활용할 수 있다. 입력되는 데이터가 크게 변하지 않는 일부 비전이나 언어 모델들은 재학습이 없어도 몇 년 동안 지속될 수도 있다. 하지만 유입되는 새로운 데이

www.kci.go.kr

터가 매번 동일하다는 보장은 없다. 코로나와 같이 예측하기 어려운 외부 요인이 발생하면 아무리 안정적으로 여겨진 데이터라도 변할 수 있다.

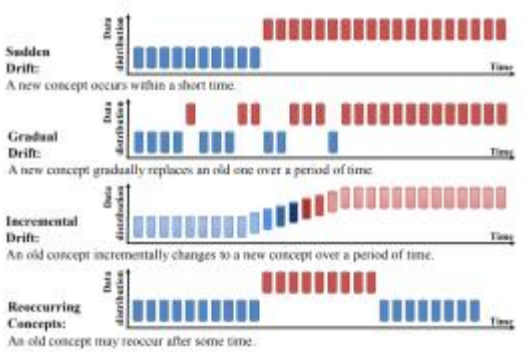

그림 1. 개념 드리프트의 종류
Fig. 1. Types of concept drift
[Source: https://arxiv.org/pdf/2004.05785.pdf]

데이터 드리프트는 공변량 이동(covariate shift)이라고도 한다. 입력 데이터의 분포가 시간에 따라 변할 때 발생하는데, 예를 들면, 연령과 소득을 기준으로 고객이 제품을 구매할 가능성을 예측하도록 훈련된 머신러닝 모델에서, 시간이 지남에 따라 고객의 연령 및 소득 분포가 크게 변하는 경우 모델은 더 이상 구매 가능성을 정확하게 예측하지 못할 수 있게 된다[1].

과거의 여러 연구에서는 지속적으로 도착하는 스트림 데이터의 맥락에서 데이터 드리프트 문제를 조사하고 이를 해결하기 위한 다양한 솔루션을 제안했다[2-5].

특성 x를 사용하여 레이블 y를 예측하는 지도 문제(supervised problem)에서 데이터 드리프트의 주요 원인은 공변량 이동이나 기능 x와 레이블 y 간의 기본 관계가 변경되는 경우로 정의한다[3]. 공변량 이동은 특성 값 x 분포의 변화를 말한다. 데이터 드리프트 문제에 대한 기존 솔루션은 창기반 방법, 이동감지 방법, 그리고 앙상블 기반 방법으로 광범위하게 분류할 수 있다. 새 모델을 훈련하기 위해 기존 데이터에 대해 슬라이딩 창을 사용하는것이 창 기반 방법이다[6]. 이동 감지 방법은 통계 테스트를 사용하여 데이터 드리프트를 감지하고 데이터 드리프트가 발생했음을 감지한 경우에만 모델을 재교육한다[7]. 앙상블 기반 방법은 이전 훈련 데이터에 대한 모델 앙상블을 훈련하고 예측의 가중 평균을 취한다[8].

드리프트의 감지(Detect) 방법은 머신러닝 모델 기반 접근과 통계적 접근방식으로 크게 구분해 볼 수 있다. 머신러닝 모델 기반 접근 방식은 들어오는 입력 데이터의 표류 여부를 감지하는 모니터링 방식이며, 통계적 접근방식은 두 확률 분포 간의 차이를 계산하여 드리프트를 탐지하는 방식으로, PSI(Population Stability Index), KL Divergence, JS Divergence, KS 테스트 및 Wasserstein Metric 등이 있다[9].

[9]에서는 머신러닝 모델의 데이터 드리프트를 감지하는 오픈 소스 활성 학습 도구 키트인 Encord Active를 사용하여 데이터 드리프트 감지하는 방법에 대해 소개하고 있다. [10]에서는 데이터 스트림의 분포가 다양한 시계열 데이터에 대한 다양한 드리프트 감지 방법을 비교 및 평가하고, 성능을 유지하기 위한 즉각적인 경고와 적절한 조치를 통해 시계열 모델의 실시간 분석에 적합한 워크플로우를 제안한다. [11]에서는 머신러닝 모델의 시계열 데이터를 위한 데이터 선정 메커니즘을 제안했던 이전 연구를 적용한 합성 데이터를 이용하여 적대적 분류기(Adversarial classifier)를 이용한 데이터 선택의 효과를 확인한다.

이외에도 [12]에서는 개념 드리프트 감지기 구성에 대한 기존 기술을 비교 분석하였고, [13]에서는 기존의 데이터 품질 관련 연구들을 비교 분석하고, 그 결과로 빅 데이터의 데이터 셋에 대한 라이프사이클을 정의하였다. [14]에서는 클라

www.kci.go.kr

우드 제공업체의 두 가지 실제 배포를 연구하여 소프트웨어 제품에서 데이터 드리프트가 미치는 영향을 기술하고 있다. 이 연구에서는 데이터 드리프트 문제에 대한 기존 솔루션은 확장성, 실측 대기 시간, 혼합 유형의 데이터 드리프트와 같은 실제 문제를 해결하도록 설계되지 않은 점을 지적하며, 데이터 드리프트 문제에 대한 확장 가능한 솔루션을 제안하고 있다. Pan의 연구에서는 이러한 현상에 대처하기 위해 정기적으로 모델을 업데이트하고, 데이터 드리프트를 유발하는 특징을 찾아 효과적으로 학습하고 모델 성능을 향상시키고자 하였다[15].

2.2 드리프트 감지 기술

일반적으로 드리프트 탐지기(Drift detector)는 학습모델의 예측 결과를 분석하고 특정 결정 모델을 적용하여 데이터 분포의 변화를 검출한다. 즉, $(\overrightarrow{x_i}, y_i)$ 형태의 샘플들이 제공되는 경우, 여기서 $\overrightarrow{x_i}$는 속성 벡터이고, y_i는 그에 대응되는 부류(class)라고 하자. 각 샘플에 대해 학습모델은 예측값($\hat{y_i}$)을 생성한다. 실제 결과(y_i)와 비교하여 예측이 올바른지($\hat{y_i} = y_i$) 아닌지($\hat{y_i} \neq y_i$) 결정한다. 이 접근 방식을 따르는 가장 잘 알려진 방법은 DDM[16], EDDM[17] 및 STEPD[18] 이다.

드리프트 탐지 방법들은 다양한 전략이나 통계를 사용하여 학습모델의 성능을 모니터링하고 개념 드리프트가 언제 발생하는지를 결정한다. 일반적으로 특정 수준보다 더 낮은 신뢰 수준이 나타나는 경우 경고를 발생시키고, 이런 경고는 개념 드리프트가 발생했음을 의미하게 된다. 여러 방법론들은 이러한 지점에서 머신러닝의 새 인스턴스를 생성하여 병렬로 훈련이 시작되도록 규정한다. 결국 개념 드리프트가 확인되면, 새로운 모델이 원래 모델을 대체하게 된다.

DDM(Drift Detection Method)은 오류율과 해당 표준 편차를 분석하여 스트림의 개념 드리프트를 감지한다. 각 위치 i에 대해 잘못된 예측을 할 확률인 오류율 p_i와 표준 편차 $s_i = \sqrt{p_i \times (1-p_i)/i}$를 정의한다. 샘플의 분포가 고정적으로 유지된다면 샘플 i의 수가 증가함에 따라 오류율 p_i가 감소해야 하므로, 오류율이 증가하면 데이터 분포에 변경이 있었고, 결과적으로 현재 기본 학습자가 오래되었다고 판단하게 된다.

EDDM(Early Drift Detection Method)은 DDM과 유사하지만 오류율이 아닌 두 개의 연속된 오류 사이의 거리를 모니터링한다. 따라서 개념이 고정되면 오류 사이의 거리가 증가하는 경향이 있으며, 감소하면 경고와 드리프트가 트리거 된다. EDDM은 점진적인 개념 드리프트를 탐지하는 데 적합하고, 갑작스러운 개념 드리프트 디텍션에는 DDM이 더 적합하다고 알려져 있다.

STEPD(Statistical test of equal proportions Detection)은 처리된 데이터들을 최근(recent)과 이전(older)으로 명명된 윈도우를 정의하여, 두 윈도우에 대해 계산된 동일한 비율의 통계 테스트를 연속성 수정과 함께 사용하는 방식이다. 이 두 윈도우에 대한 머신러닝의 정확도는 동일하다고 예상할 수 있다. 최근 윈도우의 정확도에서 상당한 차이가 감지되면 경고 및 드리프트 신호가 표시된다.

ADWIN(Adaptive Windowing)은 가변 크기의 인스턴스 슬라이딩 윈도우(W)를 사용한다. 드리프트가 감지되면 W의 크기가 줄어들고 개념이 길어질수록 W의 크기가 커진다. 동적으로 조정된 두 개의 하위 윈도우가 저장되어 이전 데이터와 최신 데이터를 나타내고, 이들 하위 윈도우의 평균 차이가 지정된 임계값 보다 높을 때 드리프트를 감지하게 된다[19].

www.kci.go.kr

PHT(Page-Hinkley Test)는 개념 드리프트 감지에도 사용되는 순차 분석 기술이다. 관찰된 값인 기본 학습자의 실제 정확도와 처리된 인스턴스까지의 평균을 계산한다. 드리프트가 발생하면 분류기가 새 인스턴스를 올바르게 분류하지 못하기 시작하여 현재 정확도와 평균 정확도가 감소하게 된다. 이 두 값 사이의 누적(U_T) 및 최소 차이(m_T)가 계산된다. U_T 값이 높을수록 관측된 값이 이전 값과 상당히 다르다는 것을 의미한다. $U_T - m_T$가 허용된 변화 크기인 지정된 임계값을 초과하면 드리프트가 감지된다. 임계값이 높을수록 거짓양성은 줄어 들지만 거짓음성이 많아지고 일부 탐지가 지연될 수 있다[20].

기존의 드리프트 탐지 연구는 주로 개념 드리프트에 관한 것으로 특히 시계열 데이터와 데이터스트림 등 특정 데이터셋에 대한 연구에 한정되어 있다. 그러나 데이터 드리프트는 시간에 따른 입력 데이터 분포의 변화를 의미하는 반면, 개념 드리프트는 모델이 예측하려는 입력 데이터와 출력 변수 간의 관계 변화를 나타내며, 머신러닝 모델을 다룰 때 이해해야 하는 다소 다른 현상이라고 할 수 있다. 또한, 데이터 드리프트는 데이터 검색과 데이터 모델링, 통계 계산을 위한 지표 마련 등이 포함되는 프로세스를 통해 이루어져야 한다[1]. 따라서 데이터 드리프트를 조기에 감지하고 적시에 개입해서 머신러닝 모델의 정확한 예측 제공이 가능하도록 반복 가능한 절차를 통한 통합적이고 체계적인 접근방법이 필요하다.

3. 2-단계 데이터 품질 평가 방법론

본 연구에서는 머신러닝 모델에서 활용하는 데이터 품질 평가 메트릭을 사용한 2-단계 데이터 품질 평가 방법론을 제안한다. 2-단계 품질평가는 머신러닝 모델을 학습하기 전에 좋은 데이터(Good data)를 얻기 위한 데이터 품질 평가와 머신러닝 모델 학습 이후에 시간이 지남에 따라 드리프트 현상 발견을 위한 데이터 품질 평가를 수행한다. 본 연구에서 제안하는 방법론은 다음의 주요 기능으로 구성되어 있다.

(1) DQM Selection (Data Quality Metric) : G_DQM & D-DQM
(2) One-Step: G-DQA
(3) Two-Step: D-DQA

머신러닝 시스템은 목적에 맞은 데이터를 활용하여 머신러닝 모델을 학습하여 구축한 후에도 데이터 변형이나 목적에 맞는 데이터가 유입되는지 지속적인 모니터링을 통해 모델의 성능 저하가 발견되는 드리프트 현상이 발생하는지 주기적으로 확인해야 한다. 머신러닝 모델의 드리프트 현상이 발생했다는 것은 데이터의 예측 분석 결과의 정확성이 떨어진다는 것을 의미한다.

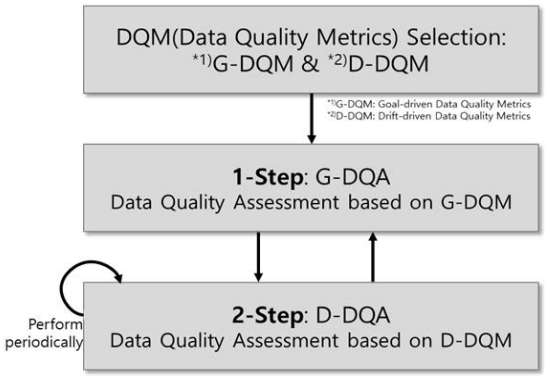

그림 2. 2-단계 데이터 품질평가
Fig. 2. Two-Steps Data quality assessment

따라서 본 연구에서 제안하는 데이터 품질 평가의 첫 번째 단계는 머신러닝 모델 구축 목표에 맞는 목적 기반 메트릭(G-DQM: Goal-driven Data Quality Metric)을 활용하여 데이터의 품질을 평가한다. 두 번째 단계는 머신러닝 모델 학

www.kci.go.kr

습 이후 시간이 지나면서 머신러닝 모델의 성능이 저하되는 드리프트 현상을 발견하기 위해 D-DQM (Drift-driven Data Quality Metric)를 활용하여 데이터의 품질을 평가하고 드리프트 유형을 확인한다. 두 번째 단계의 데이터 품질 평가는 주기적으로 수행하고 드리프트 유형에 따라 G-DQM 선정부터 다시 할지 첫 번째 단계를 반복할지 결정한다. 본 연구는 기존 연구 [21]에서 제안한 데이터 품질 평가 프레임워크를 확장하여, 그림 2와 같은 2단계 데이터 품질 평가 방법론을 제안한다.

3.1 데이터 품질 메트릭 선정

본 연구에서는 데이터 품질 메트릭을 선정하기 위해 Victor Basili가 제안한 SW 품질메트릭 선정 방법인 GQM(Goal Question Metric)을 도입하였다. 다수의 문헌과 표준에서 제시한 다양한 데이터 품질 메트릭을 목적이 다른 다양한 도메인의 머신러닝 모델에서 데이터 품질을 평가하는데 활용하는 것은 적절하지 않다. 따라서 본 연구에서는 기존에 제시된 수많은 메트릭 중에서 머신러닝 프로젝트에서 요구하는 목적에 맞게 메트릭을 선정하기 위하여 GQM 방법을 도입하였다.

3.1.1 G-DQM (Goal-Driven Data Quality Metric)

G-DQM은 머신러닝 시스템을 구축하고자 하는 목표를 설정하고 각 목표를 달성하기 위한 질문과 질문을 해결하는 단계로 메트릭을 선정한다.

G-DQM은 프로젝트 성공을 위해 달성하고자 하는 것은 무엇인지, 이를 위해 필요한 것은 무엇인지를 고려하고 빅 데이터 특성 및 머신러닝 모델을 적용하는 분야의 비즈니스 요구사항을 고려하여 목표를 설정한다.

각 목표에 대한 질문은 1:1이 될 수도 있고 1:N이 될 수도 있다. 또한 하나의 질문은 두 개의 목표를 달성할 수 있다. 따라서 목표-질문-메트릭은 다음 그림 3과 같은 관계로 표현할 수 있다.

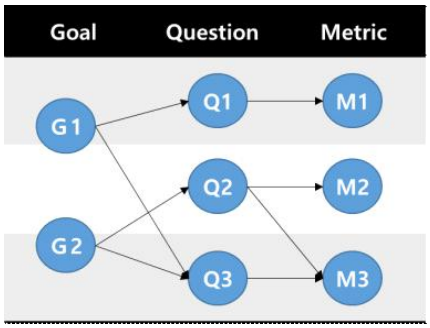

그림 3. 목표-질문-메트릭
Fig. 3. Goal-Question-Metric(GQM)

또한, 경우에 따라 목표 달성에 해당하는 메트릭을 직접적으로 매핑 할 수도 있다. 목표-(질문)-메트릭 선정 과정은 머신러닝 모델을 구축하는 목적과 용도에 적합한 메트릭을 도출할 때까지 과정을 반복한다.

메트릭을 선정하면 프로젝트에서 요구하는 품질 기준(Baseline)도 결정한다. 품질 기준은 데이터를 활용하는 목표 및 도메인에 따라 다르게 지정할 수 있다. 예를 들어, 표 1과 같이 목표와 메트릭이 정의되었다면, 국방이나 의료 등 안전성(Safety)이 매우 중요한 분야의 Accuracy 기준은 0.95로 지정할 수 있으나 다른 도메인에서는 0.9로 지정할 수도 있다.

표 1. 목표-메트릭 예시
Table 1. Goal-Metric example

Goal	Metric	Expression
데이터가 정확해야 함	Accuracy	$\frac{\text{(1-Number of data with errors)}}{\text{Total number of data}}$
중복 최소화	Redundancy	$\frac{\text{Number of duplicate data}}{\text{Total number of data}}$

www.kci.go.kr

3.1.2 D-DQM (Drift-Driven Data Quality Metric)

머신러닝 모델을 구축한 이후에는 그림 4와 같이 주기적으로 데이터 셋이 추가되므로 데이터 품질 평가도 주기적으로 수행해야 한다.

특히, 머신러닝 시스템을 운영하는 시간이 경과하면서 모델을 학습한 당시와 입력 데이터의 유형이 달라지거나 데이터와 모델의 해석 방법이 달라지는 경우 모델의 성능이 저하되어 데이터 예측의 정확도가 저하되는 드리프트 현상이 발생한다. 드리프트가 발생하면 머신러닝 모델은 새로 입력된 데이터를 사용하여 모델을 다시 학습시켜야 한다. 이렇게 모델을 업데이트해야 목적에 맞게 성능 좋은 머신러닝 시스템을 지속적으로 운영할 수 있다.

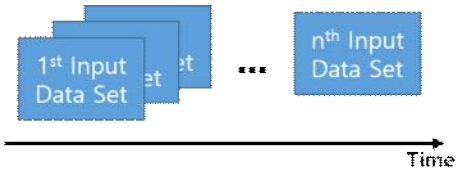

그림 4. 시간경과에 따른 데이터 셋 추가
Fig. 4. Adding data sets over time

본 연구에서는 다음 표 2와 같이 드리프트 유형에 따라 메트릭을 선정하여 주기적으로 D-DQM 기반 데이터 품질 평가를 수행한다.

표 2. 드리프트 유형별 메트릭
Table 2. Metrics by drift type

드리프트 유형	메트릭
Concept Drift(CD): Concept Drift-DQM	- Accuracy
Data Drift(DD): Data Drift-DQM	- Consistency - Completeness

개념 드리프트는 모델의 정확도를 평가하고 데이터 드리프트는 입력 데이터 패턴의 일관성 또는 완전성을 평가한다. 드리프트 메트릭은 현재 입력된 데이터를 평가하고 이전에 입력한 데이터를 비교하여 기준을 벗어나는 경우 드리프트 현상이 발생하였다고 간주한다.

3.2 2-단계 데이터 품질 평가

머신러닝 모델의 데이터 품질평가는 모델 구축 및 운영하는 전 과정에서 주기적으로 수행되어야 머신러닝 시스템의 품질이 보증된다.

본 연구에서는 기존에 저자가 제안한 목적 기반 메트릭(G-DQM)을 사용한 데이터 품질 평가 방법[21]과 드리프트 기반 메트릭(D-DQM)을 활용한 데이터 품질 평가 방법을 추가하여 2-단계 데이터 품질 평가 방법론을 제안한다. 그림 5에서 보는 것과 같이, 1-Step 데이터 품질 평가는 머신러닝 모델 학습 전에 데이터 자체를 평가하는 단계이고 2-Step 데이터 품질 평가는 머신러닝 모델 학습 이후 모델과 데이터의 관계, 현재 데이터와 이전 데이터와의 관계를 평가하는 단계이다.

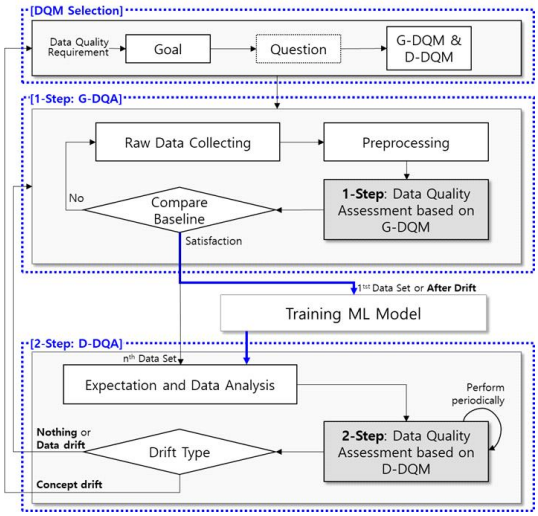

그림 5. 2-단계 데이터 품질 평가
Fig. 5. Two-Steps data quality assessment process

www.kci.go.kr

3.2.1 G-DQA: 목적기반 데이터 품질 평가

머신러닝 모델의 첫 번째 단계의 데이터 품질 평가는 최초의 머신러닝 모델을 구축하는 시점이나 머신러닝 모델을 운영하면서 개념 드리프트가 발생하는 시점에서 수행한다.

G-DQA(G-DQM based Data Quality Assessment)는 목적 기반 메트릭(G-DQM)을 사용하여 데이터 품질을 평가한다. G-DQM은 하나의 머신러닝 시스템에 여러 개를 선정할 수 있으며 G-DQM 계산식을 사용하여 데이터를 평가한 값이 품질 기준을 만족하면 다음 단계를 진행한다. G-DQA의 알고리즘이 표3에 기술되었다.

표 3. One-Step G-DQA 알고리즘
Table 3. One-Step G-DQA algorithm

Algorithm G-DQA(i, $G-DQM_i$, $Baseline_i$) :
//i: number of quality metrics
//$G-DQM_i$: metric expression
//$Baseline_i$: threshold value

for each $G-DQM_i$ where $1 \le i \le n$ **do**
  $A_i \leftarrow$ calculated $G-DQM_i$ value
  if $A_i > Baseline_i$ **then**
    print "No Satisfaction"
    Raw Data Collecting
  endif
endfor

if 1st Data set **or** After Drift **then**
  Training ML Model
else
  Data Analysis using ML Model
endif

즉, G-DQM 수식을 사용하여 데이터 품질 평가한 값 A_i은 품질 메트릭 개수(n) 만큼 산출된다. 품질 기준 $Baseline_i$과 비교하여 품질 평가 값 A_i가 $Baseline_i$ 보다 작으면, 품질 평가 값이 기준을 만족하여 데이터 품질이 확보된 것으로 간주한다. 여기에서 i는 1부터 메트릭 개수 n까지 나타낸다. 예를 들어, accuracy와 consistency의 품질 값은 각각 $A_{accuracy}$와 $A_{consistency}$이 되고, 품질 기준은 $Baseline_{accuracy}$와 $Baseline_{consistency}$로 표기한다.

데이터 품질 평가는 선정한 모든 메트릭의 평가 값이 각 품질 기준을 만족해야만 머신러닝 모델을 학습하거나 머신러닝 모델을 사용하여 예측 분석을 수행한다. 그러나 한 개라도 메트릭 평가가 품질 기준을 만족하지 못하면 새로운 데이터를 수집하고 데이터 평가를 다시 수행한다.

3.2.2 D-DQA: 드리프트기반 데이터 품질 평가

두 번째 단계의 D-DQA(D-DQM based Data Quality Assessment)는 드리프트 기반 메트릭(D- DQM)을 사용하여 데이터 품질을 평가한다.

머신러닝 시스템에서는 대량의 데이터가 주기적으로 데이터가 입력되므로 D-DQA도 주기적으로 드리프트 유형 별로 수행한다.

본 연구에서 드리프트 감지를 위해서는 드리프트 유형 별 D-DQM을 사용한 데이터를 평가한다. (n-1)번째 데이터 셋의 데이터를 평가 값과 n번째 데이터 셋의 데이터 평가 값을 비교한다. 비교한 값이 각 드리프트 메트릭의 DT(Drift Threshold)를 만족하지 못하면 드리프트 현상이 발견된 것이라 간주하고 머신러닝 모델을 새로 학습시킨다.

DT는 기존 데이터 셋과 신규로 입력된 데이터 셋의 유형 변경을 허용하는 범위이며 드리프트 메트릭마다 다르게 지정할 수 있다. 표 4의 알고리즘에 기술된 것과 같이, 예를 들어, 개념 드리프트의 메트릭을 accuracy로 선정하였다면 (n-1)번째 데이터 셋의 accuracy 평가 값 (Acc_{n-1})과 n번째 데이터 셋의 accuracy평가 값 (Acc_n)을 비교한다. 비교 값이 $DT_{Accuracy}$범위를

www.kci.go.kr

벗어나면 개념 드리프트가 발생한 것으로 간주한다. 데이터 드리프트도 마찬가지로 (n-1)번째 데이터 셋과 n번째 데이터 셋의 데이터 드리프트 메트릭 consistency 평가 값을 각각 산출하고, $DT_{Consistency}$범위 내에 있는지 확인한다.

표 4. Two-Step D-DQA 알고리즘
Table 4. Two-Step D-DQA algorithm

Algorithm D-DQA(i, DT_i) :
//$i \in (Acc, Con, Cmp)$: drift metrics
//Acc : Accuracy, Con : Consistency,
//Cmp : Completeness
// i_n : metric value for n-th data set
//DT_i: threshold value for each i

if $(Acc_{n-1} - Acc_n) > \pm DT_{Acc}$ **then**
    print "Concept Drift"
    Selection of G-DQM
else if $(Con_{n-1} - Con_n) > \pm DT_{Con}$ **or**
        $(Cmp_{n-1} - Cmp_n) > \pm DT_{Cmp}$ **then**
    print "Data Drift"
    Training ML Model in 1-Step G-DQA
endif

DT는 시스템 초기에는 임의로 정할 수 있으나 시간이 지남에 따라 평가 값이 수집이 되면 통계 시그마(σ)값을 활용하여 DT를 조정한다. 예를 들어, DT를 6-σ로 선정하는 경우 3-σ로 선정하는 시스템 보다 더 높은 데이터 품질을 요구하므로 구축하는 목적이나 도메인에 따라 DT를 결정한다.

4. 결론 및 향후 연구과제

데이터 기반 정보기술 분야에서 데이터 품질은 프로젝트의 성공 요인 중 가장 중요한 요소라고 할 수 있다. 특히 머신러닝 모델의 훈련에 사용된 데이터의 속성은 시간이 지나면서 변화하게 되는데, 이로 인해 모델의 정확도가 떨어지거나 설계된 것과 다르게 작동할 수 있게 되는 드리프트 현상이 큰 문제가 된다. 데이터 드리프트 문제는 시간 경과에 따른 머신러닝 모델의 정확도 변화 문제로, 일반적으로 모델이 훈련에 사용되는 데이터와 상당히 다른 데이터에 대해 추론을 실행하기 때문에 발생한다. 데이터 드리프트는 즉시 감지되기 어렵고, 예측이 부정확해 지기 때문에 예측을 기반으로 내린 비즈니스 결정에 어려움을 겪을 수 있다. 드리프트를 관리하기 위해 필요한 작업은 드리프트의 유형이나 범위 및 성격에 따라 달라진다. 머신러닝 모델은 변화하는 상황을 인식하지 못하기 때문에, 드리프트 문제가 발생하는 것은 단순히 데이터 뿐만 아니라 모델 자체의 문제일 수도 있다. 적절한 조치를 취하려면 드리프트 식별 뿐만 아니라 데이터 품질 관리 및 평가와 함께 드리프트 비율에 대한 임계값 설정 및 사전 경고 구성을 위한 반복 가능한 절차를 확립하는 것이 중요하다.

본 논문에서는 머신러닝 프로젝트에서 발생하는 드리프트 문제를 데이터의 품질평가 메트릭을 통해 해결해 보고자 하였다. 이를 위해 2단계 데이터 품질평가 프레임워크를 제안하고, 드리프트 탐지를 위한 평가 메트릭스와 평가 절차를 정의한다. 본 연구에서 제안하는 데이터 품질 평가의 첫 번째 단계는 머신러닝 모델 구축 목표에 맞는 목적 기반 메트릭(G-DQM)을 활용하여 데이터의 품질을 평가한다. 두 번째 단계는 머신러닝 모델 학습 이후 시간이 지나면서 머신러닝 모델의 성능의 저하되는 드리프트 현상을 발견하고 드리프트 기반 메트릭(D-DQM)을 활용하여 데이터의 품질을 평가하고 드리프트 유형을 확인한다. 두 번째 단계의 데이터 품질평가는 주기적으로 수행하고 드리프트 유형에 따라 G-DQM 선정부터 다시 할지 첫 번째 단계를 반복할지 결정

www.kci.go.kr

하게 된다.

본 논문에서 정의한 2단계 데이터 품질 평가 프레임워크는 데이터 드리프트의 문제를 데이터 품질관리 및 평가의 차원에서 반복가능한 절차를 통해 접근해 보았다. 현재까지 데이터 드리프트 문제에 대한 모범사례(best-practice)가 없는 상황에서, 지속적인 모니터링 체계 확립을 통해 이를 해결해 보고자 하였다.

향후 제안된 프레임워크 구현을 통해, 실제 데이터셋의 품질 관리 사례를 마련하는 것이 필요하며, 기존 연구들과의 비교 평가를 통한 유효성 검증이 필요하다.

이 논문은 정부(과학기술정보통신부)의
재원으로 정보통신기획평가원의 지원을 받아
수행된 지역지능화혁신인재양성사업임
(IITP-2024- RS-2022-00156334)

참 고 문 헌

[1] M. Ali, “Understanding Data Drift and Model Drift : Drift Detection in Python”, Founder&Creator of PyCaret, Jan. 2023, https://www.datacamp.com/tutorial/understanding-data-drift-model-drift

[2] A. Bifet, R. Gavaldà, “Learning from time-changing data with adaptive windowing”, In Proceedings of the SIAM International Conference on Data Mining (ICDM), pp. 443-448, 2007, DOI: https://doi.org/10.1137/1.9781611972771.42

[3] J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N. V. Chawla, F. Herrera, “A unifying view on dataset shift in classification”, Pattern recognition, 45(1), pp. 521-530, 2012, DOI: https://doi.org/10.1016/j.patcog.2011.06.019

[4] A. Suprem, J. Arulraj, C. Pu, J. Ferreira, “ODIN: Automated drift detection and recovery in video analytics”, In Proceedings of the VLDB Endowent, 13(12), pp. 2453-2465, July 2020, DOI: https://doi.org/10.48550/arXiv.2009.05440

[5] A. Tahmasbi, E. Jothimurugesan, S. Tirthapura, P. B. Gibbons, “Driftsurf: A risk-competitive learning algorithm under concept drift”, ArXiv journal, 2020, DOI : https://doi.org/10.48550/arXiv.2003.06508

[6] G. Widmer, M. Kubat, “Learning in the presence of concept drift and hidden contexts”, Machine learning, 23(1), pp. 69-101, 1996, DOI: https://doi.org/10.1007/BF00116900

[7] A. Pesaranghader, H. L. Viktor, E. Paquet, “Mcdiarmid drift detection methods for evolving data streams”, In Proceedings of the International Joint Conference on Neural Networks, pp.1-9, 2018, DOI: https://doi.org/10.1109/IJCNN.2018.8489260

[8] D. Brzezinski, J. Stefanowski, “Reacting to different types of concept drift: The accuracy updated ensemble algorithm”, IEEE Transactions on Neural Networks and Learning Systems, 25(1), pp. 81-94, 2014, https://doi.org/10.1109/TNNLS.2013.2251352

[9] A. Acharya, “How to Detect Data Drift on Datasets”, August 9, 2023, available at https://encord.com/blog/detect-data-drift/

[10] S. Ashok, S. Ezhumalai, T. Patwa, “Remediating data drifts and re-establishing ML models”, In Proceedings of International Conference on Machine Learning and Data Engineering, 218, pp. 799-809, 2023, DOI: https://doi.org/10.1016/j.procs.2023.01.060

[11] Y. Konno, M. Nakano, M. Oguchi, “Efficient Data Selection Indicators for Updating Models under Data Drifted Environment”, In Proceedings of International Conference on Big Data, pp. 6724-6726, 2022, DOI: https://doi.org/

www.kci.go.kr

10.1109/BigData55660.2022.10020630
[12] R. S. Barros, S. Garrido T. Carvalho Santos, “A large-scale comparison of concept drift detectors”, Information Science, vol.451-452, pp.348-370, 2018, DOI: https://doi.org/10.1016/j.ins.2018.04.014
[13] Y. Gong, G. Liu, Y. Xue, R. Li, L. Meng, “A Survey on dataset quality in machine learning”, Infromation and software technology, 162(107268), 2023, DOI: https://doi.org/10.1016/j.infsof.2023.107268
[14] A. Mallick, K. Hsieh, B. Arzani, G. Joshi, “Matchmaker: Data Drift mitigation in machine learning for large-scale systems”, In Proceedings of the Machine Learning and Systems, pp. 77-94, 2022,
[15] J. Pan, V. Pham, M. Dorairaj, H. Chen, J. Lee, “Adversarial validation approach to concept drift problem in automated machine learning systems”, journal of ArXiv, 2020, https://doi.org/10.48550/arXiv.2004.03045
[16] J. Gama , P. Medas , G. Castillo , P. Rodrigues, “Learning with drift detection”, LNAI, vol. 3171, pp. 286-295, 2004, DOI: https://doi.org/10.1007/978-3-540-28645-5_29
[17] M. Baena-Garcia, J. del Campo-Ávila, R. Fidalgo, A. Bifet, R. Gavald, R. Morales-Bueno, “Early drift detection method”, In Proceedings of the International Workshop on Knowledge Discovery from Data Streams, pp. 77-86, 2006,
[18] K. Nishida, K. Yamauchi, “Detecting concept drift using statistical testing”, LNCS, vol. 4755, pp. 264-269, 2007, DOI: https://doi.org/10.1007/978-3-540-75488-6_27
[19] A. Bifet , R. Gavaldà, “Learning from time-changing data with adaptive windowing”, In Proceedings of SIAM International Conference on Data Mining, pp. 443-448, 2007, DOI: https://doi.org/10.1137/1.9781611972771.42
[20] J. Gama, “Knowledge Discovery from Data Streams”, Chapman & Hall/CRC Data mining and knowledge discovery series, 2010
[21] O. Choi, Y. Kim, “A Survey of Data Quality Assessment Methods for Big Data”, Journal of Software Assessment and Valuation, 19(4), pp. 89-98, 2023, DOI : http://dx.doi.org/10.29056/jsav.2023.12.09

저 자 소 개

최옥주(Okjoo Choi)

2008.2 숙명여자대학교 컴퓨터과학과 박사
1990.8-1996.3 LG생산기술원 주임원구원
1996.7-2009.8 한국오라클 수석컨설턴트
2009.9-2022.12 한국과학기술원 연구교수
2023.10-2024.2 강원대학교 산학협력중점교수
2024.03-현재 배재대학교 조교수
<주관심분야> 빅데이터 분석, 데이터마이닝, 데이터 품질, 소프트웨어 품질, 프로젝트 관리

김유경(Yukyong Kim)

2001.8 숙명여자대학교 컴퓨터과학과 박사
2005.9-2006.8 UC Davis, Post-doc.
2006.9-2013.9 한양대학교 컴퓨터공학과 연구교수
2018.3-현재 숙명여자대학교 기초공학부 교수
<주관심분야> 웹서비스 QoS 평가, SOA 기반 IoT 신뢰 평가, 소프트웨어 품질평가

www.kci.go.kr