

인공지능 시스템 품질 평가 방안

기능 정확성, 기능 적응성, 강건성, 투명성 중심으로

포멀웍스 정세훈 책임

2024.06.12

목차

I. 발표를 시작하며

II. 인공지능 시스템 품질 평가 방안 개요

III. 인공지능 시스템 품질 평가 방안

IV. 결론 및 요약

I. 발표를 시작하며

1. 5년차 소프트웨어 V&V 실무자 입장에서 본 인공지능 소프트웨어 시스템 품질 평가

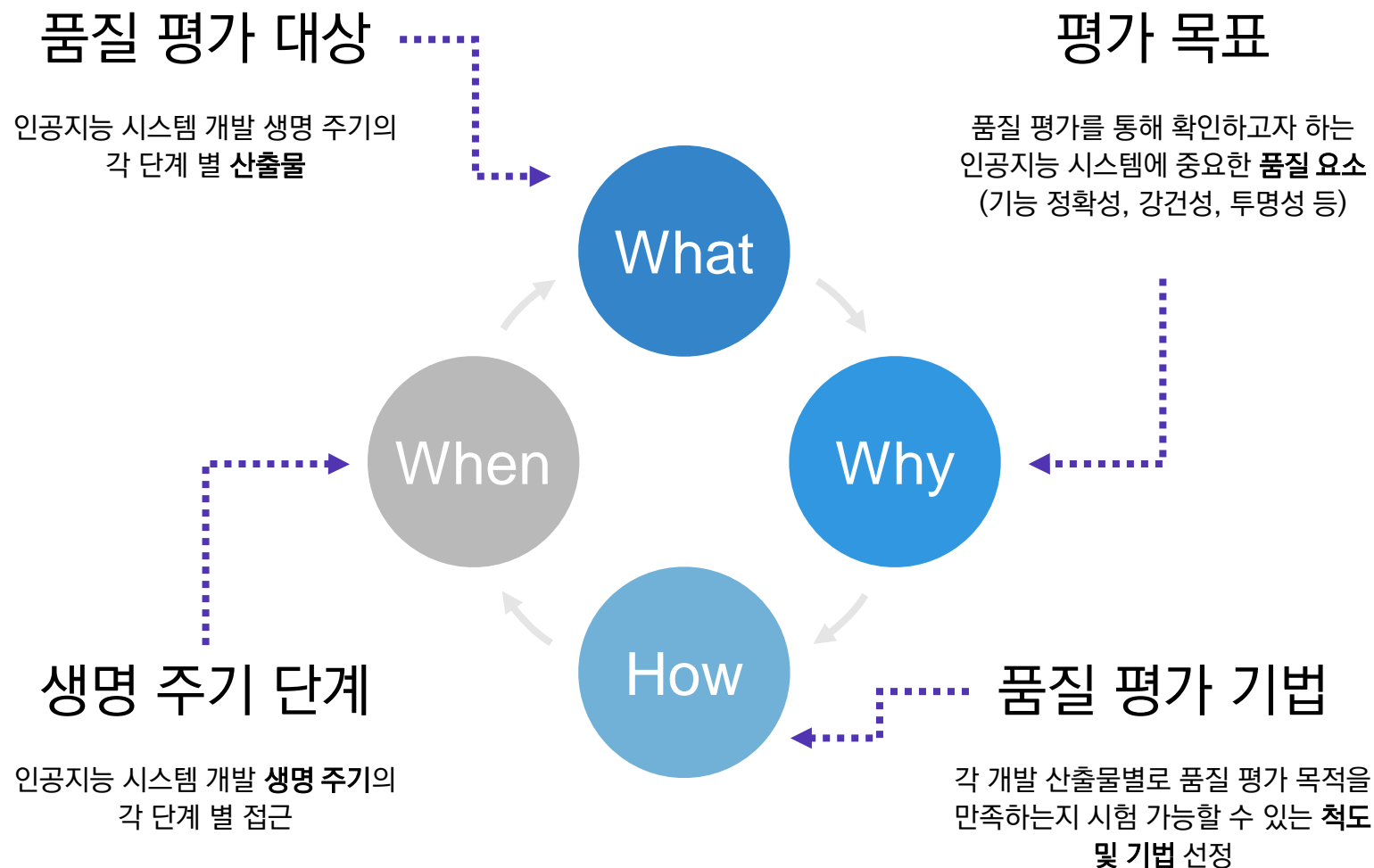
- 전통적인 소프트웨어 시스템
 - 코드가 진실 (전체가 아닐지라도)
 - ❖ “Truth can only be found in one place: the code.”, Robert C. Martin (“Clean Code”의 저자)
 - ❖ “the code is truth, but not the whole truth.”, Grady Booch (UML의 창시자)
 - 소프트웨어 V&V를 수행할 때 SRS/코드 리뷰와 시스템 시험을 통해 어느 정도의 “확신”을 가질 수 있음.
- 인공지능 시스템 (특히 기계 학습 시스템)
 - 코드나 데이터 조작 반쪽짜리 진실
 - ❖ “What is not in the data cannot be learned. What is in the data is *likely learned*, but *not always perfectly*.”, ISO/IEC TR 5469:2024 Functional safety and AI systems
- 소프트웨어 V&V를 수행할 때 최종적인 인공지능 시스템의 성능 뿐만 아니라 개발 과정에 대한 종합적인 평가도 중요하다고 생각함.

소프트웨어 V&V (Verification & Validation) ?

- 소프트웨어가 올바른 과정으로 만들어 졌는지 확인 (Are you building it right?)
- 만든 소프트웨어가 올바른 지 확인 (Are you building the right thing?)

II. 인공지능 시스템 품질 평가 방안 개요

1. 인공지능 소프트웨어 시스템 품질 평가 방안 수립을 위한 네 가지 고려 사항

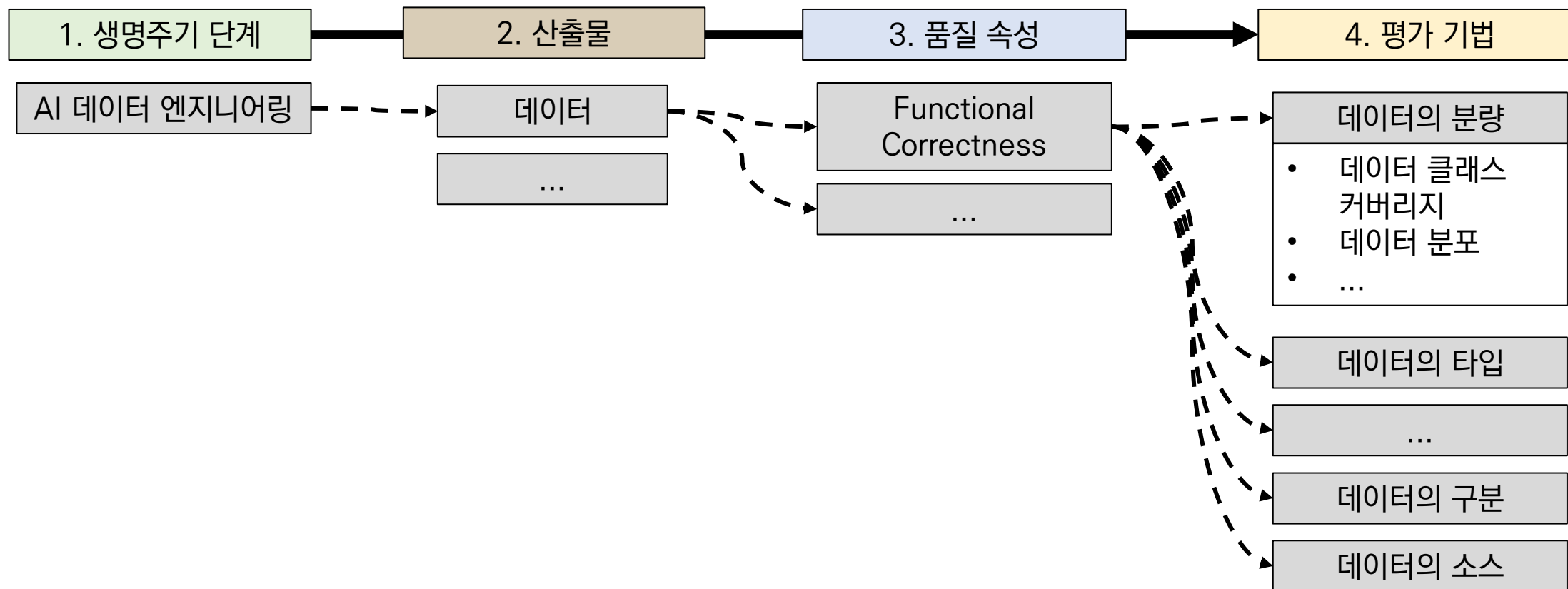


II. 인공지능 시스템 품질 평가 방안 개요

2. 인공지능 소프트웨어 시스템 품질 평가에 관한 네 가지 고려 사항의 연계

➤ 인공지능 소프트웨어 시스템 개발 생명주기의 각 스테이지 별로 품질 평가 방안 분석

● [예] AI 데이터 엔지니어링 스테이지 분석 예시



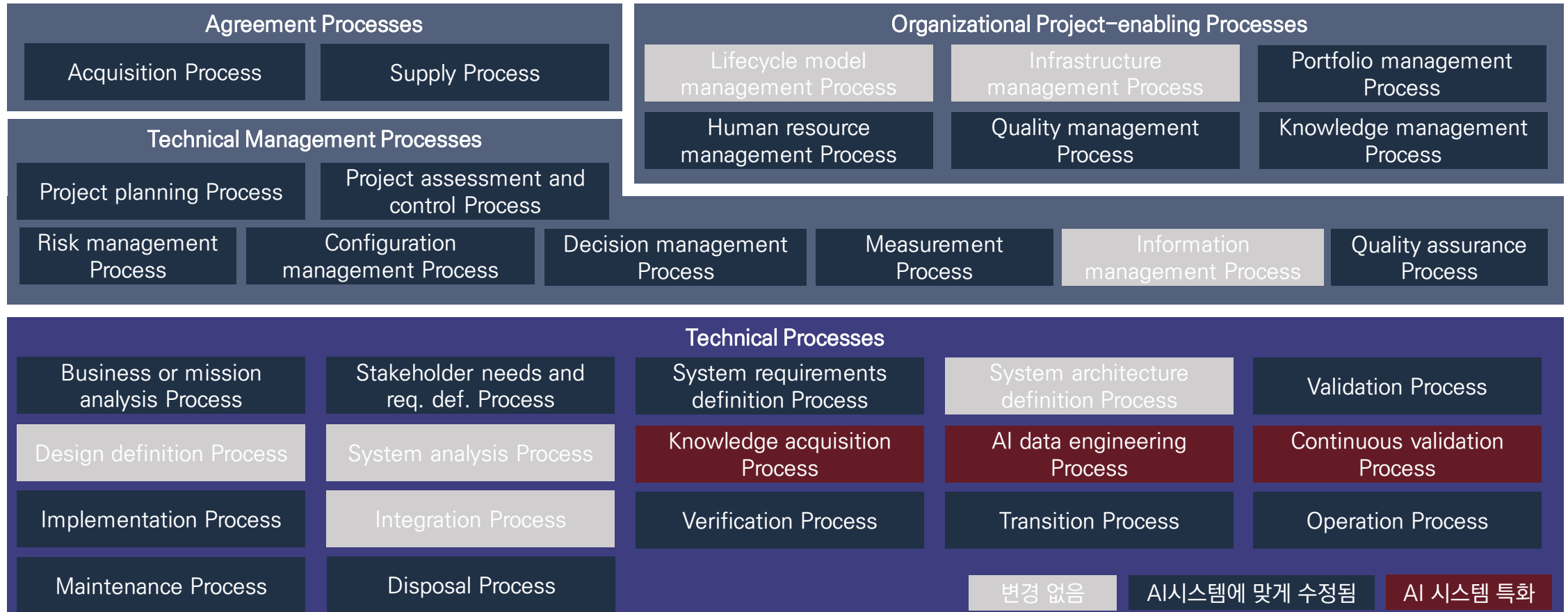
III. 인공지능 시스템 품질 평가 방안

1-1. 인공지능 소프트웨어 시스템 개발 생명 주기

➤ 인공지능 소프트웨어 시스템 개발 생명주기 참조 모델

● ISO/IEC 5338:2023 AI System life cycle processes

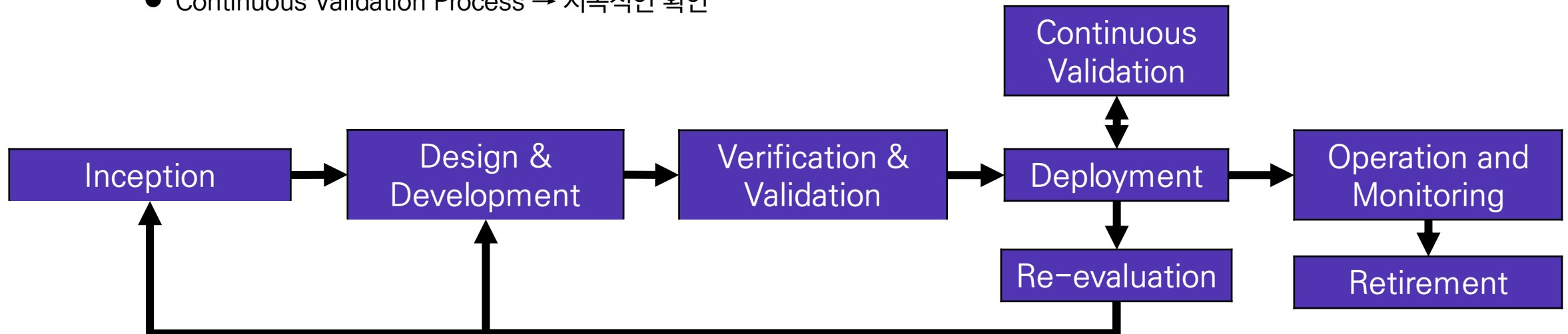
❖ ISO/IEC/IEEE 15288:2023 System life cycle process 기반



III. 인공지능 시스템 품질 평가 방안

1-2. 인공지능 소프트웨어 시스템 개발 생명 주기

- Technical Processes는 인공지능 소프트웨어 시스템 실제 개발과 관련된 프로세스들
 - 세부 프로세스는 입/출력 산출물이 명시된 “스테이지”들로 구성
- 인공지능 소프트웨어 시스템 개발에 중요한 프로세스 및 스테이지가 존재
 - Inception Process → *비즈니스 및 미션 분석, 이해당사자 요구사항 정의*, 시스템 요구사항 정의
 - Design & Development Process → *시스템 아키텍처 정의, 설계 정의, 시스템 분석*, 도메인 지식 확보, AI 데이터 엔지니어링, 구현, 통합
 - Verification & Validation Process → 확인 및 검증
 - Continuous Validation Process → 지속적인 확인



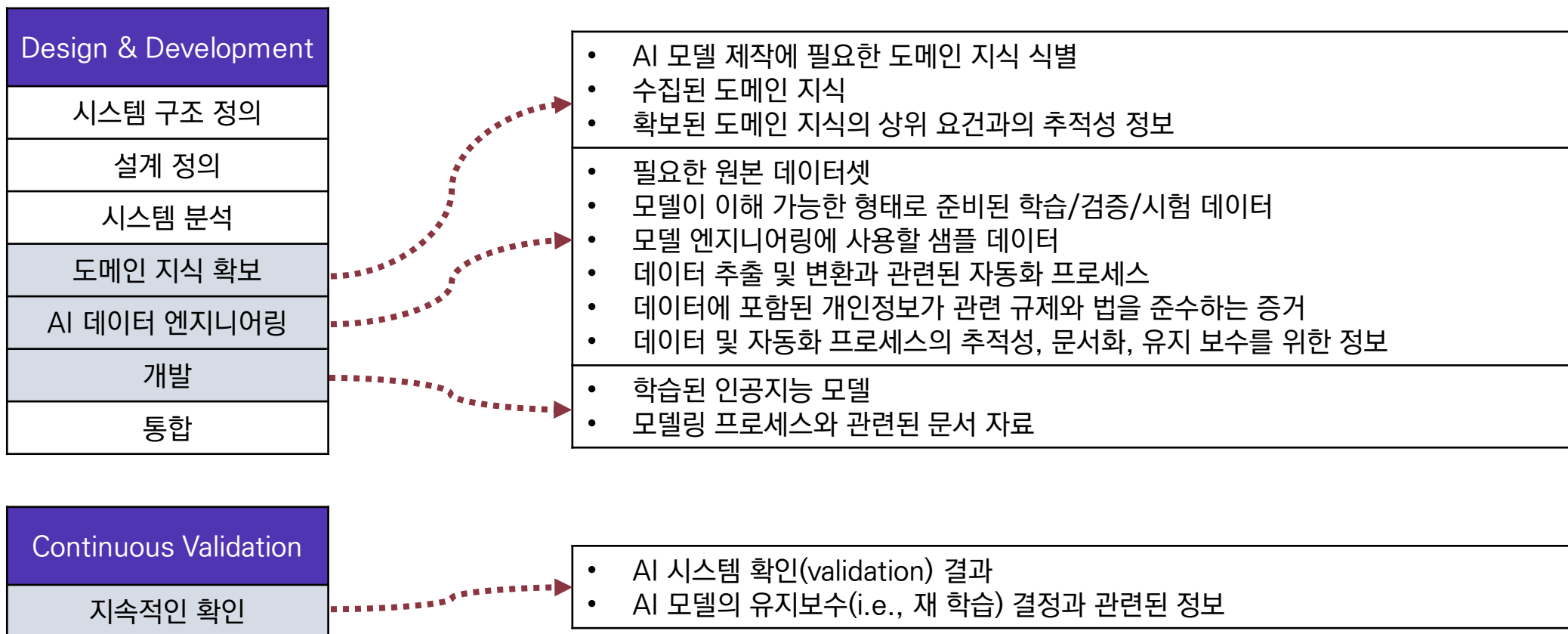
III. 인공지능 시스템 품질 평가 방안

2. 인공지능 소프트웨어 시스템 품질 평가 범위

➤ Technical Processes 생명주기의 각 산출물을 평가

● [예] Technical Processes

❖ Design & Development Process 및 Continuous Validation Process 산출물



III. 인공지능 시스템 품질 평가 방안

3. 인공지능 소프트웨어 시스템 품질 평가 속성

- 기존 소프트웨어 시스템 대비 인공지능 소프트웨어 시스템 특화 품질 속성
 - ISO/IEC 25059: 2023, “Quality model for AI systems”
 - 기능 정확성, 강건성, 기능 적응성, 투명성, 제어 가능성, 개입 가능성

- 기존 소프트웨어 시스템 대비 인공지능 소프트웨어 시스템 안전을 위한 고려 속성
 - ISO/IEC TR 5469: 2024, “Functional safety and AI systems”
 - 자율성, 제어, 투명성, 설명가능성, 명세가능성, 운영 환경의 변화, 적응성, 강건성

AI 시스템 특화 품질 속성 ISO/IEC 25059: 2023	AI 시스템 안전 고려 속성 ISO/IEC 5469: 2024
5.2 제어 가능성 (Controllability)	8.2 자율성과 제어성 (Level of automation and control)
5.3 기능 적응성 (Functional Adaptability)	8.3 투명성과 설명가능성 (Degree of transparency and explainability)
5.4 기능 정확성 (Functional Correctness)	8.4.1 복잡한 환경과 모호한 명세 (Complexity of the environment and vague specifications)
5.5 강건성 (Robustness)	8.4.2.1 데이터 변화 (Data drift)
5.6 투명성 (Transparency)	8.4.2.2 컨셉 변화 (Concept drift)
5.7 개입 가능성(Intervenability)	8.4.3 환경에 적응 (Issues related to learning from environment)
-	8.5 강건성 (Resilience to adversarial and intentional malicious inputs)

III. 인공지능 시스템 품질 평가 방안

3. 인공지능 소프트웨어 시스템 품질 평가 속성

- 기존 소프트웨어 시스템 대비 인공지능 소프트웨어 시스템 특화 품질 속성
 - ISO/IEC 25059: 2023, “Quality model for AI systems”
 - 기능 정확성, 강건성, 기능 적응성, 투명성, 제어 가능성, 개입 가능성

- 기존 소프트웨어 시스템 대비 인공지능 소프트웨어 시스템 안전을 위한 고려 속성
 - ISO/IEC TR 5469: 2024, “Functional safety and AI systems”
 - 자율성, 제어, 투명성, 설명가능성, 명세가능성, 운영 환경의 변화, 적응성, 강건성

AI 시스템 품질 요소	품질 요소 설명
[A] 기능 정확성 (Functional Correctness)	학습된 AI 모델의 결과(예측)의 정확도
[B] 강건성 (Robustness)	학습된 AI 모델의 성능이 만족스러움을 유지할 수 있는 동요의 범위
[C] 기능 적응성 (Functional Adaptability)	AI 시스템이 요구되는 성능을 유지할 수 있는 운영 환경 변화의 정도
[D] 투명성 (Transparency)	AI 시스템이 사용하는 알고리즘과 데이터에 대한 접근성

III. 인공지능 시스템 품질 평가 방안

4-1. 인공지능 소프트웨어 시스템 품질 평가 기법: 기능 정확성

➤ 프로젝트나 시스템이 올바른 결과를 요구되는 정밀도로 제공하는 정도

프로세스	산출물	기능 정확성 품질 평가 방안
구현 프로세스	학습된 인공지능 모델	<ul style="list-style-type: none"> • 기능 정확성 시험: 시스템 요구사항에서 지정된 기능 정확성 측정 척도를 활용해 학습된 AI 모델이 상위 요건에서 요구하는 기능 정확성 수준을 확보했는지 시험 <ul style="list-style-type: none"> • Precision, Recall, F-1 score, MSE, AUC, ... • 이전 버전과 비교: 이전 버전의 인공지능 모델 혹은 전통적인 소프트웨어 시스템과 비교하여 향상된 기능 정확성 수준을 보이는지 시험 <ul style="list-style-type: none"> ➤ 비교를 통해 보다 구체적인 기능 정확성 평가가 가능 • 용례 기반 시험: 학습된 AI 모델이 다양한 입력 데이터에 대해 일관된 기능 정확성을 보이는지 시험 <ul style="list-style-type: none"> ➤ 다양한 사용 시나리오에 대해 구체적인 기능 정확성 평가가 가능 • 교차 검증: 선택한 AI 모델이 다양한 학습 데이터에 대해 일관된 기능 정확성을 보이는지 시험 <ul style="list-style-type: none"> ➤ 다양한 학습 환경에 대해 구체적인 기능 정확성 평가가 가능
	모델링 프로세스와 관련된 문서 자료	<ul style="list-style-type: none"> • 구현 프로세스 검증: 모델링 과정을 포함한 구현 절차에 대한 문서를 검토하여 완성된 AI 모델 및 시스템이 재현성을 보장할 수 있는 수준으로 작성되었으며, 실제 산출물도 해당 프로세스를 통해 구현 되었는지 검토. <ul style="list-style-type: none"> ✓ 확보된 AI 모델 및 시스템의 기능 정확성이 우연이 아님을 보임. • 추적성 검증: 구현 프로세스의 각 과정이 상위 설계나 요구사항과 일관되는지 검토 <ul style="list-style-type: none"> ✓ AI 모델 및 시스템의 기능 정확도를 확보하는 과정이 우연이 아님을 보임.

III. 인공지능 시스템 품질 평가 방안

4-1. 인공지능 소프트웨어 시스템 품질 평가 기법: 기능 정확성

➤ 프로젝트나 시스템이 올바른 결과를 요구되는 정밀도로 제공하는 정도

프로세스	산출물	기능 정확성 품질 평가 방안
도메인 지식 확보 프로세스	AI 모델 제작에 필요한 도메인 지식 식별	<ul style="list-style-type: none"> • 도메인 지식 평가: 정확한 도메인 지식이 충분히 수집되었는지 확인하기 위한 체크리스트 작성 및 검토. 체크리스트는 확보된 도메인 지식에 문제 정의, 주요 개념, 데이터 소스, 관련 규제가 포함되어 있는지 작성되어야 함. <ul style="list-style-type: none"> ✓ 도메인 지식은 시스템 요구사항과 모델 엔지니어링과 AI 데이터 엔지니어링을 연결하는 주요 자료로, 요구되는 기능 정확도를 확보하는데 필수적. • 도메인 지식 계보 검토: 문서화 된 도메인 지식 항목의 출처 및 획득 방법 검토 <ul style="list-style-type: none"> ➤ 지식의 출처를 통해 지식의 신뢰성을 간접적으로 확인하여 신뢰할 수 없는 지식이 기능 적 확성에 미칠 수 있는 악영향을 방지. • 도메인 지식 획득과정 검토: 문서화 된 도메인 지식 획득 과정 검토 <ul style="list-style-type: none"> ✓ 관련성: 이후 단계에서 기능 정확성 확보에 문제 발생 시 도메인 지식 수준에서의 원인을 추적하고 수정할 수 있음.
	수집된 도메인 지식	<ul style="list-style-type: none"> • 저장 형식 및 무결성 검토: 도메인 지식이 표준화된 저장 형식으로 쓰기 편하게 저장되어 있으며 의도치 않은 변조를 탐지 및 방지하고 있는지 검토. <ul style="list-style-type: none"> ✓ 도메인 지식의 낮은 사용성과 변조된 지식으로 인한 모델 엔지니어링 및 AI 데이터 엔지니어링 오류 및 비효율성을 방지. • 버전 관리: 저장된 지식에 대한 버전 관리 시스템이 적용여부 검토. <ul style="list-style-type: none"> ✓ 잘못된 버전의 지식을 사용으로 인한 기능 오류 방지.
	확보된 도메인 지식의 상위 요건과의 추적성 정보	<ul style="list-style-type: none"> • 요구사항 추적성 검토: 도메인 지식이 AI 모델의 기능적 요구사항과 일치하는지 검토. <ul style="list-style-type: none"> ✓ 수집된 지식이 모델의 요구사항과 일치하는지 확인함으로써 요구사항에 어울리지 않는 지식에서 파생된 모델과 데이터가 기능 정확성을 떨어뜨리지 않도록 해야 함.

III. 인공지능 시스템 품질 평가 방안

4-2. 인공지능 소프트웨어 시스템 품질 평가 기법: 강건성

➤ 시스템이 보지 못했거나, 편향되거나, 악의적 혹은 잘못된 입력이 존재 하더라도 기능 정확도를 유지할 수 있는 정도

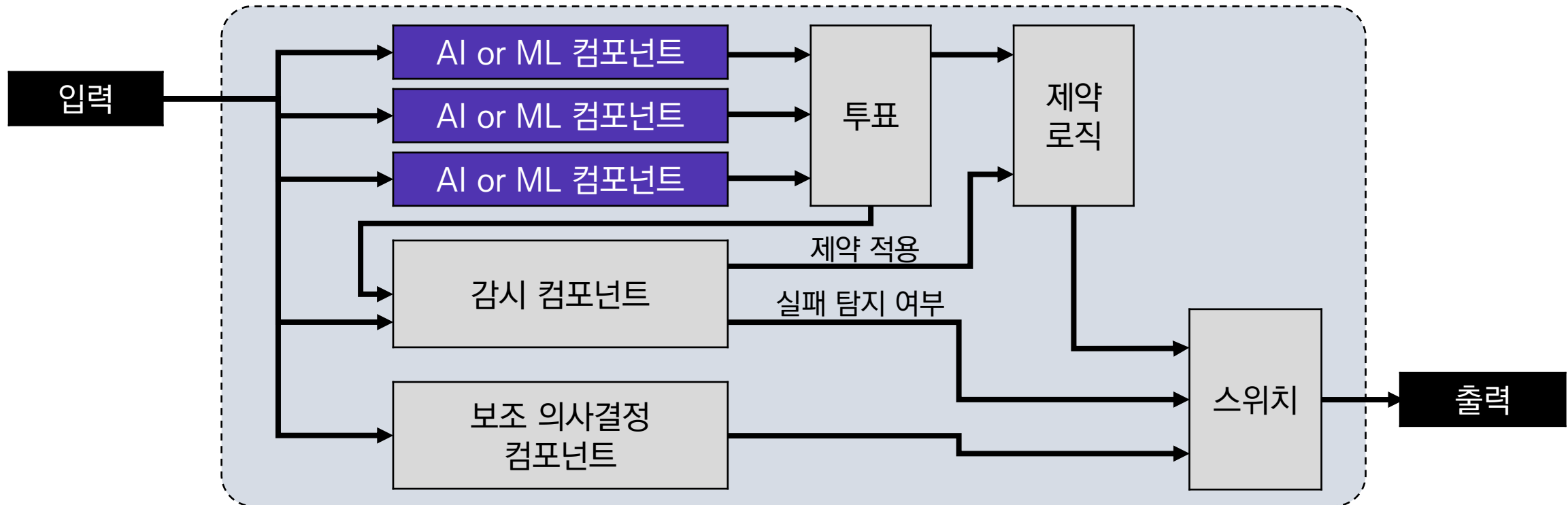
프로세스	산출물	강건성 품질 평가 방안
AI 데이터 엔지니어링 프로세스	모델이 이해 가능한 형태로 준비된 학습/검증/시험 데이터	<ul style="list-style-type: none"> • 데이터 다양성 평가: 데이터 셋이 여러 소스, 다양한 환경, 다양한 경계 사례를 반영한 데이터인지 평가 <ul style="list-style-type: none"> ✓ 다양한 데이터는 AI 모델이 실제 환경에서 안정적으로 작동하는데 도움이 됨.
	데이터 추출 및 변환과 관련된 자동화 프로세	<ul style="list-style-type: none"> • 데이터 클리닝 프로세스 평가: 데이터에서 outlier와 노이즈를 식별하고 처리하는 절차를 확인 <ul style="list-style-type: none"> ✓ Outliner와 노이즈가 학습된 AI 모델의 일반화 능력에 주는 악영향을 방지
구현 프로세스	학습된 인공지능 모델	<ul style="list-style-type: none"> • 다양한 데이터 조건에서 성능 평가: 미리 정의된 수준 내에서 노이즈가 포함된 데이터, 리소스 제약 아래에서 악의적으로 생성된 입력 데이터 등 에서도 성능을 유지하는지 시험 <ul style="list-style-type: none"> ✓ 다양한 데이터는 AI 모델이 실제 환경에서 안정적으로 작동하는데 도움이 됨. • 일관성 평가: 동일한 입력 데이터에 대해 학습된 AI 모델이 일관된 결과를 제공하는지 확인 <ul style="list-style-type: none"> ✓ 확보된 강건성이 우연이 아님을 보임.
	모델링 프로세스와 관련된 문서 자료	<ul style="list-style-type: none"> • 구현 프로세스 검증: 모델링 과정을 포함한 구현 절차에 대한 문서를 검토하여 완성된 AI 모델 및 시스템이 재현성을 보장할 수 있는 수준으로 작성되었으며, 실제 산출물도 해당 프로세스를 통해 구현 되었는지 검토. <ul style="list-style-type: none"> ✓ 확보된 AI 모델 및 시스템의 강건성이 우연이 아님을 보임. • 추적성 검증: 구현 프로세스의 각 과정이 상위 설계나 요구사항과 일관되는지 검토 <ul style="list-style-type: none"> ✓ AI 모델 및 시스템의 강건성을 확보하는 과정이 우연이 아님을 보임.

III. 인공지능 시스템 품질 평가 방안

4-2. 인공지능 소프트웨어 시스템 품질 평가 기법: 강건성

➤ 디자인 리뷰를 통한 강건성 확인

- 다양한 AI 모델 출력을 참조하는 설계
- 잘못된 입력(i.e., 범위에서 벗어난 입력) 탐지 및 정해진 범위의 출력으로 제약하는 설계
- 보조 의사결정 컴포넌트 (i.e., 정확도가 떨어지지만 전통적인 소프트웨어로 작성)가 존재하는 설계



III. 인공지능 시스템 품질 평가 방안

4-2. 인공지능 소프트웨어 시스템 품질 평가 기법: 강건성

- 악의적 입력 데이터 생성 기법(i.e., gradient based adversarial attack) 활용으로 강건성 강화
 - 생성된 악의적 입력 데이터를 학습 데이터로 재 활용
- White-box 시험을 통한 강건성 확인
 - Coverage를 달성하기 위한 시험 사례를 생성하여 학습된 모델의 행위 강건성 시험
 - ❖ 전통적인 시험 관점에서 직관적이거나, Coverage 달성 과정에서 관찰된 모델의 행위가 올바른지 판단이 필요하며 Coverage와 시험 사례 품질 사이의 관계성이 아직 불투명*

Deep Neural Network 종류	Coverage 예시	설명
Convolutional Neural Network (CNN)	Neuron coverage (SOSP '17)	시험 사례에 의해 t 초과의 activation value를 가진 적이 있는 뉴런의 비율 (e.g., Line coverage)
	Sign-sign coverage (TECS '19)	시험 사례에 의해 인접한 뉴런의 부호가 바뀌며 자신의 부호가 바뀐 뉴런의 비율 (e.g., MCDC)
Recurrent Neural Network (RNN 혹은 LSTM)	Hidden state coverage (TSE '22)	시험 사례에 의해 최대값이 된 적 있는 RNN 셀(cell)들의 비율
	Cell state coverage (TSE '22)	시험 사례에 의해 특정 범위(ex: -1.0 ~ -0.25)가 된 LSTM RNN의 Cell state의 비율

* S. Yan et.al, "Correlations between deep neural network model coverage criteria and model quality," ESEC/FSE 2020

III. 인공지능 시스템 품질 평가 방안

4-3. 인공지능 소프트웨어 시스템 품질 평가 기법: 기능 적응성

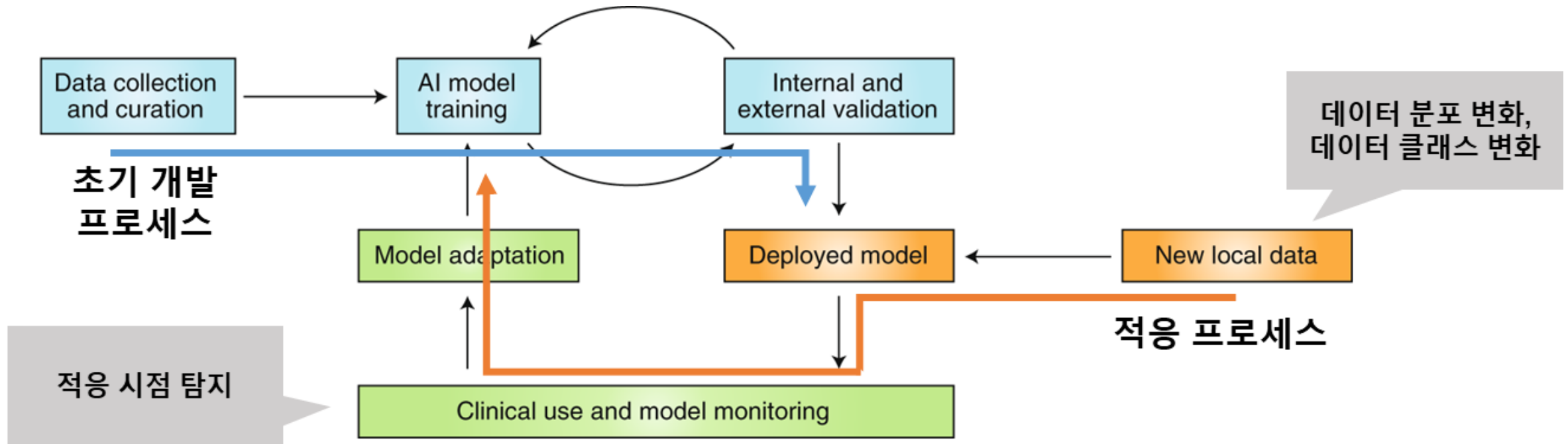
➤ 프로젝트나 시스템이 설치된 환경의 변화에 적응하는 정도

프로세스	산출물	기능 적응성 품질 평가 방안
도메인 지식 확보 프로세스	AI 모델 제작에 필요한 도메인 지식 식별	<ul style="list-style-type: none"> • 도메인 지식 평가: 식별된 도메인 지식이 AI 소프트웨어 시스템의 작동 환경에서 발생할 수 있는 data drift 혹은 concept drift에 관한 내용을 포함하는지 확인. <ul style="list-style-type: none"> ✓ 예상할 수 있는 data drift, concept drift를 기반으로 AI 데이터 엔지니어링 및 모델 엔지니어링 단계에서 기능 적응성을 확보.
	수집된 도메인 지식	<ul style="list-style-type: none"> • 도메인 지식 저장 방식 평가: 식별된 도메인 지식을 저장하는 형태를 평가하여 새로운 정보나 변화에 따라 쉽게 업데이트 될 수 있는지 확인. <ul style="list-style-type: none"> ✓ AI 소프트웨어 시스템의 효율적인 기능 적응성 확보 및 시험을 위해 새로운 도메인 지식이 쉽게 갱신 수 있어야 함. • 도메인 지식 갱신 프로세스 평가: 식별된 도메인 지식이 AI 소프트웨어 시스템의 작동 환경에서 발생할 수 있는 data drift 혹은 concept drift에 관한 내용을 포함하는지 확인. <ul style="list-style-type: none"> ✓ AI 소프트웨어 시스템의 효율적인 기능 적응성 확보 및 시험을 위해 새로운 도메인 지식이 체계적으로 관리되어야 함.
구현 프로세스	학습된 인공지능 모델	<ul style="list-style-type: none"> • 연속 학습 및 모델 갱신 기능 평가: AI 소프트웨어 시스템이 필요한 기능 적응성을 갖추고 있는지 시험. <ul style="list-style-type: none"> ✓ 기능 적응성 구현에 필요한 적응 시점 탐지, 재학습, 회귀 시험, 비정상 상황 기록 및 보고 기능이 올바르게 구현되어야 함.
	모델링 프로세스와 관련된 문서 자료	<ul style="list-style-type: none"> • 연속 학습 및 모델 갱신 프로세스 평가: AI 소프트웨어 시스템의 기능 적응성 관련 프로세스 평가. <ul style="list-style-type: none"> ✓ 기능 적응성 구현에 필요한 적응 시점 탐지, 재학습, 회귀 시험, 비정상 상황 기록 및 보고 기능에 관한 명세가 올바르게 완전하며 일관되게 문서화 되어야 함.

III. 인공지능 시스템 품질 평가 방안

4-3. 인공지능 소프트웨어 시스템 품질 평가 기법: 기능 적응성

- 기능 적응성이 고려된 개발/운용 프로세스 예시



<https://www.nature.com/articles/s42256-020-0185-2>

III. 인공지능 시스템 품질 평가 방안

4-4. 인공지능 소프트웨어 시스템 품질 평가 기법: 투명성

➤ AI 시스템 개발 프로세스에 관한 데이터의 가용성과 제3자 검토가 가능한 정도

프로세스	산출물	투명성 품질 평가 방안
도메인 지식 확보 프로세스	AI 모델 제작에 필요한 도메인 지식 식별	<ul style="list-style-type: none"> • 지식 식별 프로세스 문서화: 도메인 지식 식별 과정이 문서화되어 있는지 확인 및 각 단계에서 식별된 지식의 출처와 그 이유가 명확히 기록되어 있는지 확인. <ul style="list-style-type: none"> ✓ 지식 식별 과정을 투명하게 문서화하여 이해 당사자들이 지식이 어떻게 선택되었고 왜 중요한지 이해할 수 있게 함. • 도메인 전문가 검토 기록: 도메인 전문가가 참여하여 지식을 식별한 과정과 그 검토 결과가 기록되어 있는지 확인. <ul style="list-style-type: none"> ✓ 전문가 검토 과정을 명확하게 기록하여 신뢰성을 확보함.
	수집된 도메인 지식	<ul style="list-style-type: none"> • 지식 저장소 접근성 검사: 저장된 지식이 쉽게 접근 가능하고 필요한 이해 관계자들에게 적절히 제공되는지 확인하며, 접근 권한과 데이터의 가용성을 평가. <ul style="list-style-type: none"> ✓ 지식이 쉽게 접근 가능하면 이해 관계자들이 필요할 때 정보를 확인하고 사용할 수 있어 투명성이 높아집니다. • 지식 저장 구조와 메타데이터 검토: 저장된 지식이 체계적으로 구조화되어 있고, 메타데이터(각 지식 항목의 출처, 획득 방법, 시기)가 충분히 기록되어 있는지 확인. <ul style="list-style-type: none"> ✓ 체계적인 구조와 풍부한 메타데이터는 지식의 추적성과 이해를 돕고, 시스템의 투명성을 높이는 데 기여함.
	확보된 도메인 지식의 상위 요건과의 추적성 정보	<ul style="list-style-type: none"> • 추적성 검증: 각 지식 항목의 출처와 획득 방법이 명확하게 기록되어 있는지 확인. <ul style="list-style-type: none"> ✓ 지식의 신뢰성과 정확성 검증을 가능케 하여 투명성을 확보 • 지식 갱신 및 변경 기록 검토: 지식이 갱신되거나 변경된 경우 그 기록이 이유와 함께 명확히 남아 있는지 확인. <ul style="list-style-type: none"> ✓ 지식 갱신과 관련된 모든 정보가 명확하게 추적 가능해져 투명성을 확보.

III. 인공지능 시스템 품질 평가 방안

4-4. 인공지능 소프트웨어 시스템 품질 평가 기법: 투명성

➤ AI 시스템 개발 프로세스에 관한 데이터의 가용성과 제3자 검토가 가능한 정도

프로세스	산출물	투명성 품질 평가 방안
AI 데이터 엔지니어링 프로세스	모델이 이해 가능한 형태로 준비된 학습/검증/시험 데이터	<ul style="list-style-type: none"> • 데이터 출처 및 처리 과정 문서화 검토: 훈련 데이터 및 검증 데이터의 출처와 데이터 처리 과정이 명확하게 문서화되어 있는지 확인. 데이터 수집, 전 처리, 포맷 변환 과정 등이 상세히 기록되어 있는지 평가. <ul style="list-style-type: none"> ✓ 데이터 출처와 처리 과정을 이해 관계자들이 쉽게 확인할 수 있음. • 테스트 데이터 검증 기록 검토: 테스트 데이터가 다양한 시나리오와 조건을 반영하는지 확인. 테스트 데이터의 검증 과정과 결과가 상세히 기록되어 있는지 평가. <ul style="list-style-type: none"> ✓ 테스트 데이터가 다양한 조건을 반영하고 검증 과정을 명확히 기록하여 테스트 결과의 신뢰성을 이해당사자들이 쉽게 확인할 수 있게 함.
	모델 엔지니어링에 사용할 샘플 데이터	<ul style="list-style-type: none"> • 수동 분석 데이터 문서화: 수동 분석에 사용된 데이터의 출처와 처리 과정 문서 검토. 수동 분석의 목적과 방법, 결과가 투명하게 기록되어 있는지 평가. <ul style="list-style-type: none"> ✓ 이해 관계자들의 분석 결과에 대한 신뢰성과 이해도를 높일 수 있음. • 수동 분석 결과 검토: 수동 분석의 결과가 명확히 기록되고, 이를 통해 얻어진 인사이트가 적절히 반영되었는지 검토. <ul style="list-style-type: none"> ✓ 모델 엔지니어링 과정을 설명하여 투명성을 높일 수 있음.
	데이터 추출 및 변환과 관련된 자동화 프로세스	<ul style="list-style-type: none"> • 데이터 추출/변환/로드 프로세스 문서화 검토: 각 과정의 각 단계와 사용된 도구, 방법 등이 투명하게 기록되어 있는지 평가. <ul style="list-style-type: none"> ✓ 데이터 처리 과정의 투명성을 확보할 수 있음.

III. 인공지능 시스템 품질 평가 방안

4-4. 인공지능 소프트웨어 시스템 품질 평가 기법: 투명성

➤ “신뢰할 수 있는 인공지능 개발 안내서”의 투명성 평가 체크리스트 발췌

개발 단계	대상 AI 컴포넌트	투명성 평가 항목	적용 이유
Inception Process	학습모델	AI 시스템 생명주기에 걸쳐 나타날 수 있는 위험 요소를 분석하였는가?	기능 오동작(예: 오인식)에 따른 위험 요소를 분석해야 함.
	학습모델	위험요소를 제어 및 방지하거나 영향을 완화하기 위한 방안을 마련하였는가?	분석된 위험요소를 제거하거나 파급효과를 감소시키기 위한 방안을 마련해야 함.
	학습모델, 데이터	AI 시스템의 특성을 고려한 테스트 환경을 설계하였는가?	테스트 데이터의 통계적 특성은 시험 검증 대상의 실제 운영 환경과 유사해야 함.
	학습모델, 데이터	AI 시스템의 테스트 설계에 필요한 협의 체계를 구성하였는가?	효과적인 시험 검증을 위해 시험 검증 대상 개발 조직과 검증 조직의 협의 체계를 구성해야 함.
Design and Development Process: 도메인 지식 확보 프로세스, AI 데이터 엔지니어링 프로세스	데이터, 프레임워크	데이터의 명확한 이해와 활용을 지원하는 상세 정보를 제공하는가?	데이터, 메타데이터, 전처리 과정이 투명해야 함.
	데이터	데이터의 출처는 기록 및 관리되고 있는가?	데이터의 출처 혹은 추출 방식을 신뢰할 수 있어야 함.
	데이터	학습에 사용되는 특성을 분석하고 선정 기준을 마련하였는가?	선정된 데이터의 특성(feature)이 시험 검증 대상 구현에 적합한지 설명 가능해야 함.
	데이터	데이터 라벨링 시, 발생 가능한 편향을 확인하고 방지하였는가?	데이터의 레이블은 일관된 기준으로 작성되어야 함.
	데이터	데이터의 편향 방지를 위한 샘플링을 수행하였는가?	학습/검증/시험 데이터는 임의로 샘플링하여 데이터의 편향을 방지해야 함.
Design and Development Process: 구현 프로세스	프레임워크	오픈소스 라이브러리의 안정성을 확인하였는가?	신뢰할 수 있는 개발 프레임워크를 사용해야 함.
	프레임워크	오픈소스 라이브러리의 위험 요소는 관리되고 있는가?	개발 프레임워크의 라이선스 관계를 확인해야 하며, 알려진 보안 취약점을 관리해야 함.
	학습모델	사용자가 모델 추론 결과의 도출 과정을 수용할 수 있도록 근거를 제공하는가?	가급적이면 모델의 추론 결과를 설명할 수 있어야 함.
	학습모델	AI 모델 상세 문서를 통해 모델의 명세를 투명하게 제공하는가?	가급적이면 모델의 추론 결과를 설명할 수 있어야 함.
	학습모델	AI 모델 추론 결과에 대한 설명을 제공하는가?	가급적이면 모델의 추론 결과를 설명할 수 있어야 함.

IV. 결론 및 요약

1. 결론 및 요약

- 인공지능 소프트웨어 시스템 품질 평가가 중요해 지고 있음.
 - 전통적인 소프트웨어 시스템과 달리 코드에 대한 테스트로는 충분하지 않음
- 인공지능 소프트웨어 시스템 개발 과정을 기반으로 체계적인 접근이 중요하다고 생각함
 - 인공지능 소프트웨어 시스템 개발 생명 주기의 각 산출물 별로,
 - 인공지능 소프트웨어 시스템에 중요한 품질 속성 (i.e., 기능 정확성/적응성, 강건성, 투명성)을 확인하기 위한
 - 타당성 있는 시험 검증 기법 식별 및 적용.

2. 참고 문헌

- ISO/IEC 5338:2023, AI system life cycle processes
- ISO/IEC/IEEE 12207:2017, Software life cycle processes
- ISO/IEC 25059:2023, Quality model for AI systems
- ISO/IEC TR 29119-11, Guidelines on the testing of AI-based systems
- “신뢰할 수 있는 인공지능 개발 안내서”, 2022, 한국정보통신기술협회

부록

- A. 기능안전성 품질 평가 기법
- B. 기능적응성 품질 평가 기법
- C. 강건성 품질 평가 기법
- D. 투명성 품질 평가 기법

인공지능 소프트웨어 시스템 품질 평가 기법: [A] 기능 정확성

➤ 프로젝트나 시스템이 올바른 결과를 요구되는 정밀도로 제공하는 정도

프로세스	산출물	기능 정확성 품질 평가 방안
도메인 지식 확보 프로세스	AI 모델 제작에 필요한 도메인 지식 식별	<ul style="list-style-type: none"> • 도메인 지식 평가: 정확한 도메인 지식이 충분히 수집되었는지 확인하기 위한 체크리스트 작성 및 검토. 체크리스트는 확보된 도메인 지식에 문제 정의, 주요 개념, 데이터 소스, 관련 규제가 포함되어 있는지 작성되어야 함. <ul style="list-style-type: none"> ✓ 도메인 지식은 시스템 요구사항과 모델 엔지니어링과 AI 데이터 엔지니어링을 연결하는 주요 자료로, 요구되는 기능 정확도를 확보하는데 필수적. • 도메인 지식 계보 검토: 문서화 된 도메인 지식 항목의 출처 및 획득 방법 검토 <ul style="list-style-type: none"> ➤ 지식의 출처를 통해 지식의 신뢰성을 간접적으로 확인하여 신뢰할 수 없는 지식이 기능 적 확성에 미칠 수 있는 악영향을 방지. • 도메인 지식 획득과정 검토: 문서화 된 도메인 지식 획득 과정 검토 <ul style="list-style-type: none"> ✓ 관련성: 이후 단계에서 기능 정확성 확보에 문제 발생 시 도메인 지식 수준에서의 원인을 추적하고 수정할 수 있음.
	수집된 도메인 지식	<ul style="list-style-type: none"> • 저장 형식 및 무결성 검토: 도메인 지식이 표준화된 저장 형식으로 쓰기 편하게 저장되어 있으며 의도치 않은 변조를 탐지 및 방지하고 있는지 검토. <ul style="list-style-type: none"> ✓ 도메인 지식의 낮은 사용성과 변조된 지식으로 인한 모델 엔지니어링 및 AI 데이터 엔지니어링 오류 및 비효율성을 방지. • 버전 관리: 저장된 지식에 대한 버전 관리 시스템이 적용여부 검토. <ul style="list-style-type: none"> ✓ 잘못된 버전의 지식을 사용으로 인한 기능 오류 방지.
	확보된 도메인 지식의 상위 요건과의 추적성 정보	<ul style="list-style-type: none"> • 요구사항 추적성 검토: 도메인 지식이 AI 모델의 기능적 요구사항과 일치하는지 검토. <ul style="list-style-type: none"> ✓ 수집된 지식이 모델의 요구사항과 일치하는지 확인함으로써 요구사항에 어울리지 않는 지식에서 파생된 모델과 데이터가 기능 정확성을 떨어뜨리지 않도록 해야 함.

인공지능 소프트웨어 시스템 품질 평가 기법: [A] 기능 정확성

➤ 프로젝트나 시스템이 올바른 결과를 요구되는 정밀도로 제공하는 정도

프로세스	산출물	기능 정확성 품질 평가 방안
AI 데이터 엔지니어링 프로세스	모델이 이해 가능한 형태로 준비된 학습/검증/시험 데이터	<ul style="list-style-type: none"> • 데이터 품질 평가: 데이터의 3C(Correctness, Consistency, Completeness)를 평가하며, 시험 데이터의 경우 각 시험 사례의 대표성을 평가 <ul style="list-style-type: none"> ✓ 학습 데이터 품질은 학습된 AI 모델의 기능 정확성과 직접적으로 영향을 주며, 대표성이 부족한 시험 데이터는 기능 정확성 시험 결과의 신뢰성을 떨어트림. • 데이터 포맷 검토: 데이터의 포맷이 AI 모델이 가정하고 있는 타입과 일치하는지 확인. <ul style="list-style-type: none"> ➤ 잘못된 데이터 변환으로 인한 기능 정확성 문제를 방지.
	데이터 추출 및 변환과 관련된 자동화 프로세스	<ul style="list-style-type: none"> • 데이터 추출/변환/로드 프로세스 검증: 해당 프로세스가 올바르게 작동하여 원본 데이터로부터 AI 데이터가 손실없이 정확히 처리되는지 확인 <ul style="list-style-type: none"> ✓ 추출/변환/로드 과정에서 데이터가 부정확해 지거나 누락이 발생하면 기능 정확성에 악영향을 줄 수 있음.
	데이터에 포함된 개인 정보가 관련 규제와 법을 준수하는 증거	<ul style="list-style-type: none"> • 법적 준수 여부 검토: 수집/저장/처리하는 프로세스가 관련 규제와 법을 준수하며, 사용해서는 안 되는 데이터가 학습 및 평가에 포함되지 않았는지 검토 <ul style="list-style-type: none"> ✓ 사용해서는 안 되는 데이터를 통해 기능 정확성이 확보되는 것을 방지
	데이터 및 자동화 프로세스의 추적성, 문서화, 유지 보수를 위한 정보	<ul style="list-style-type: none"> • AI 데이터 문서화 검토: 데이터의 출처, 처리 이력, 변환 과정 등이 명확히 문서화되어 있는지 확인. <ul style="list-style-type: none"> ✓ 관련성: AI 데이터 생성 과정을 추적하여 이후 모델 엔지니어링 단계에서 기능적 정확성 확보를 도울 수 있음. • 형상 관리: AI 데이터의 형상 관리 프로세스를 평가 <ul style="list-style-type: none"> ✓ 관련성: 체계적인 형상 관리를 통해 잘못되거나 최신이 아닌 AI 데이터가 이후 단계에 사용되어 기능 정확성을 저해하는 것을 방지.

인공지능 소프트웨어 시스템 품질 평가 기법: [A] 기능 정확성

➤ 프로젝트나 시스템이 올바른 결과를 요구되는 정밀도로 제공하는 정도

프로세스	산출물	기능 정확성 품질 평가 방안
구현 프로세스	학습된 인공지능 모델	<ul style="list-style-type: none"> • 기능 정확성 시험: 시스템 요구사항에서 지정된 기능 정확성 측정 척도를 활용해 학습된 AI 모델이 상위 요건에서 요구하는 기능 정확성 수준을 확보했는지 시험 <ul style="list-style-type: none"> • Precision, Recall, F-1 score, MSE, AUC, ... • 이전 버전과 비교: 이전 버전의 인공지능 모델 혹은 전통적인 소프트웨어 시스템과 비교하여 향상된 기능 정확성 수준을 보이는지 시험 <ul style="list-style-type: none"> ➤ 비교를 통해 보다 구체적인 기능 정확성 평가가 가능 • 용례 기반 시험: 학습된 AI 모델이 다양한 입력 데이터에 대해 일관된 기능 정확성을 보이는지 시험 <ul style="list-style-type: none"> ➤ 다양한 사용 시나리오에 대해 구체적인 기능 정확성 평가가 가능 • 교차 검증: 선택한 AI 모델이 다양한 학습 데이터에 대해 일관된 기능 정확성을 보이는지 시험 <ul style="list-style-type: none"> ➤ 다양한 학습 환경에 대해 구체적인 기능 정확성 평가가 가능
	모델링 프로세스와 관련된 문서 자료	<ul style="list-style-type: none"> • 구현 프로세스 검증: 모델링 과정을 포함한 구현 절차에 대한 문서를 검토하여 완성된 AI 모델 및 시스템이 재현성을 보장할 수 있는 수준으로 작성되었으며, 실제 산출물도 해당 프로세스를 통해 구현 되었는지 검토. <ul style="list-style-type: none"> ✓ 확보된 AI 모델 및 시스템의 기능 정확성이 우연이 아님을 보임. • 추적성 검증: 구현 프로세스의 각 과정이 상위 설계나 요구사항과 일관되는지 검토 <ul style="list-style-type: none"> ✓ AI 모델 및 시스템의 기능 정확도를 확보하는 과정이 우연이 아님을 보임.

인공지능 소프트웨어 시스템 품질 평가 기법: [B] 기능 적응성

➤ 프로젝트나 시스템이 설치된 환경의 변화에 적응하는 정도

프로세스	산출물	기능 적응성 품질 평가 방안
도메인 지식 확보 프로세스	AI 모델 제작에 필요한 도메인 지식 식별	<ul style="list-style-type: none"> • 도메인 지식 평가: 식별된 도메인 지식이 AI 소프트웨어 시스템의 작동 환경에서 발생할 수 있는 data drift 혹은 concept drift에 관한 내용을 포함하는지 확인. <ul style="list-style-type: none"> ✓ 예상할 수 있는 data drift, concept drift를 기반으로 AI 데이터 엔지니어링 및 모델 엔지니어링 단계에서 기능 적응성을 확보.
	수집된 도메인 지식	<ul style="list-style-type: none"> • 도메인 지식 저장 방식 평가: 식별된 도메인 지식을 저장하는 형태를 평가하여 새로운 정보나 변화에 따라 쉽게 업데이트 될 수 있는지 확인. <ul style="list-style-type: none"> ✓ AI 소프트웨어 시스템의 효율적인 기능 적응성 확보 및 시험을 위해 새로운 도메인 지식이 쉽게 갱신 수 있어야 함. • 도메인 지식 갱신 프로세스 평가: 식별된 도메인 지식이 AI 소프트웨어 시스템의 작동 환경에서 발생할 수 있는 data drift 혹은 concept drift에 관한 내용을 포함하는지 확인. <ul style="list-style-type: none"> ✓ AI 소프트웨어 시스템의 효율적인 기능 적응성 확보 및 시험을 위해 새로운 도메인 지식이 체계적으로 관리되어야 함.
구현 프로세스	학습된 인공지능 모델	<ul style="list-style-type: none"> • 연속 학습 및 모델 갱신 기능 평가: AI 소프트웨어 시스템이 필요한 기능 적응성을 갖추고 있는지 시험. <ul style="list-style-type: none"> ✓ 기능 적응성 구현에 필요한 적응 시점 탐지, 재학습, 회귀 시험, 비정상 상황 기록 및 보고 기능이 올바르게 구현되어야 함.
	모델링 프로세스와 관련된 문서 자료	<ul style="list-style-type: none"> • 연속 학습 및 모델 갱신 프로세스 평가: AI 소프트웨어 시스템의 기능 적응성 관련 프로세스 평가. <ul style="list-style-type: none"> ✓ 기능 적응성 구현에 필요한 적응 시점 탐지, 재학습, 회귀 시험, 비정상 상황 기록 및 보고 기능에 관한 명세가 올바르게 완전하며 일관되게 문서화 되어야 함.

인공지능 소프트웨어 시스템 품질 평가 기법: [C] 강건성

➤ 시스템이 보지 못했거나, 편향되거나, 악의적 혹은 잘못된 입력이 존재 하더라도 기능 정확도를 유지할 수 있는 정도

프로세스	산출물	강건성 품질 평가 방안
AI 데이터 엔지니어링 프로세스	모델이 이해 가능한 형태로 준비된 학습/검증/시험 데이터	<ul style="list-style-type: none"> • 데이터 다양성 평가: 데이터 셋이 여러 소스, 다양한 환경, 다양한 경계 사례를 반영한 데이터인지 평가 <ul style="list-style-type: none"> ✓ 다양한 데이터는 AI 모델이 실제 환경에서 안정적으로 작동하는데 도움이 됨. • 데이터 클리닝 프로세스 평가: 데이터에서 outlier와 노이즈를 식별하고 처리하는 절차를 확인 <ul style="list-style-type: none"> ✓ Outliner와 노이즈가 학습된 AI 모델의 일반화 능력에 주는 악영향을 방지
구현 프로세스	학습된 인공지능 모델	<ul style="list-style-type: none"> • 다양한 데이터 조건에서 성능 평가: 미리 정의된 수준 내에서 노이즈가 포함된 데이터, 리소스 제약 아래에서 악의적으로 생성된 입력 데이터 등 에서도 성능을 유지하는지 시험 <ul style="list-style-type: none"> ✓ 다양한 데이터는 AI 모델이 실제 환경에서 안정적으로 작동하는데 도움이 됨. • 일관성 평가: 동일한 입력 데이터에 대해 학습된 AI 모델이 일관된 결과를 제공하는지 확인 <ul style="list-style-type: none"> ✓ 확보된 강건성이 우연이 아님을 보임.
	모델링 프로세스와 관련된 문서 자료	<ul style="list-style-type: none"> • 구현 프로세스 검증: 모델링 과정을 포함한 구현 절차에 대한 문서를 검토하여 완성된 AI 모델 및 시스템이 재현성을 보장할 수 있는 수준으로 작성되었으며, 실제 산출물도 해당 프로세스를 통해 구현 되었는지 검토. <ul style="list-style-type: none"> ✓ 확보된 AI 모델 및 시스템의 강건성이 우연이 아님을 보임. • 추적성 검증: 구현 프로세스의 각 과정이 상위 설계나 요구사항과 일관되는지 검토 <ul style="list-style-type: none"> ✓ AI 모델 및 시스템의 강건성을 확보하는 과정이 우연이 아님을 보임.

인공지능 소프트웨어 시스템 품질 평가 기법: [D] 투명성

➤ AI 시스템 개발 프로세스에 관한 데이터의 가용성과 제3자 검토가 가능한 정도

프로세스	산출물	투명성 품질 평가 방안
도메인 지식 확보 프로세스	AI 모델 제작에 필요한 도메인 지식 식별	<ul style="list-style-type: none"> • 지식 식별 프로세스 문서화: 도메인 지식 식별 과정이 문서화되어 있는지 확인 및 각 단계에서 식별된 지식의 출처와 그 이유가 명확히 기록되어 있는지 확인. <ul style="list-style-type: none"> ✓ 지식 식별 과정을 투명하게 문서화하여 이해 당사자들이 지식이 어떻게 선택되었고 왜 중요한지 이해할 수 있게 함. • 도메인 전문가 검토 기록: 도메인 전문가가 참여하여 지식을 식별한 과정과 그 검토 결과가 기록되어 있는지 확인. <ul style="list-style-type: none"> ✓ 전문가 검토 과정을 명확하게 기록하여 신뢰성을 확보함.
	수집된 도메인 지식	<ul style="list-style-type: none"> • 지식 저장소 접근성 검사: 저장된 지식이 쉽게 접근 가능하고 필요한 이해 관계자들에게 적절히 제공되는지 확인하며, 접근 권한과 데이터의 가용성을 평가. <ul style="list-style-type: none"> ✓ 지식이 쉽게 접근 가능하면 이해 관계자들이 필요할 때 정보를 확인하고 사용할 수 있어 투명성이 높아집니다. • 지식 저장 구조와 메타데이터 검토: 저장된 지식이 체계적으로 구조화되어 있고, 메타데이터(각 지식 항목의 출처, 획득 방법, 시기)가 충분히 기록되어 있는지 확인. <ul style="list-style-type: none"> ✓ 체계적인 구조와 풍부한 메타데이터는 지식의 추적성과 이해를 돕고, 시스템의 투명성을 높이는 데 기여함.
	확보된 도메인 지식의 상위 요건과의 추적성 정보	<ul style="list-style-type: none"> • 추적성 검증: 각 지식 항목의 출처와 획득 방법이 명확하게 기록되어 있는지 확인. <ul style="list-style-type: none"> ✓ 지식의 신뢰성과 정확성 검증을 가능케 하여 투명성을 확보 • 지식 갱신 및 변경 기록 검토: 지식이 갱신되거나 변경된 경우 그 기록이 이유와 함께 명확히 남아 있는지 확인. <ul style="list-style-type: none"> ✓ 지식 갱신과 관련된 모든 정보가 명확하게 추적 가능해져 투명성을 확보.

인공지능 소프트웨어 시스템 품질 평가 기법: [D] 투명성

➤ AI 시스템 개발 프로세스에 관한 데이터의 가용성과 제3자 검토가 가능한 정도

프로세스	산출물	투명성 품질 평가 방안
AI 데이터 엔지니어링 프로세스	모델이 이해 가능한 형태로 준비된 학습/검증/시험 데이터	<ul style="list-style-type: none"> • 데이터 출처 및 처리 과정 문서화 검토: 훈련 데이터 및 검증 데이터의 출처와 데이터 처리 과정이 명확하게 문서화되어 있는지 확인. 데이터 수집, 전 처리, 포맷 변환 과정 등이 상세히 기록되어 있는지 평가. <ul style="list-style-type: none"> ✓ 데이터 출처와 처리 과정을 이해 관계자들이 쉽게 확인할 수 있음. • 테스트 데이터 검증 기록 검토: 테스트 데이터가 다양한 시나리오와 조건을 반영하는지 확인. 테스트 데이터의 검증 과정과 결과가 상세히 기록되어 있는지 평가. <ul style="list-style-type: none"> ✓ 테스트 데이터가 다양한 조건을 반영하고 검증 과정을 명확히 기록하여 테스트 결과의 신뢰성을 이해당사자들이 쉽게 확인할 수 있게 함.
	모델 엔지니어링에 사용할 샘플 데이터	<ul style="list-style-type: none"> • 수동 분석 데이터 문서화: 수동 분석에 사용된 데이터의 출처와 처리 과정 문서 검토. 수동 분석의 목적과 방법, 결과가 투명하게 기록되어 있는지 평가. <ul style="list-style-type: none"> ✓ 이해 관계자들의 분석 결과에 대한 신뢰성과 이해도를 높일 수 있음. • 수동 분석 결과 검토: 수동 분석의 결과가 명확히 기록되고, 이를 통해 얻어진 인사이트가 적절히 반영되었는지 검토. <ul style="list-style-type: none"> ✓ 모델 엔지니어링 과정을 설명하여 투명성을 높일 수 있음.
	데이터 추출 및 변환과 관련된 자동화 프로세스	<ul style="list-style-type: none"> • 데이터 추출/변환/로드 프로세스 문서화 검토: 각 과정의 각 단계와 사용된 도구, 방법 등이 투명하게 기록되어 있는지 평가. <ul style="list-style-type: none"> ✓ 데이터 처리 과정의 투명성을 확보할 수 있음.

인공지능 소프트웨어 시스템 품질 평가 기법: [D] 투명성

➤ “신뢰할 수 있는 인공지능 개발 안내서”의 투명성 평가 체크리스트 발췌

개발 단계	대상 AI 컴포넌트	투명성 평가 항목	적용 이유
Inception Process	학습모델	AI 시스템 생명주기에 걸쳐 나타날 수 있는 위험 요소를 분석하였는가?	기능 오동작(예: 오인식)에 따른 위험 요소를 분석해야 함.
	학습모델	위험요소를 제어 및 방지하거나 영향을 완화하기 위한 방안을 마련하였는가?	분석된 위험요소를 제거하거나 파급효과를 감소시키기 위한 방안을 마련해야 함.
	학습모델, 데이터	AI 시스템의 특성을 고려한 테스트 환경을 설계하였는가?	테스트 데이터의 통계적 특성은 시험 검증 대상의 실제 운영 환경과 유사해야 함.
	학습모델, 데이터	AI 시스템의 테스트 설계에 필요한 협의 체계를 구성하였는가?	효과적인 시험 검증을 위해 시험 검증 대상 개발 조직과 검증 조직의 협의 체계를 구성해야 함.
Design and Development Process: 도메인 지식 확보 프로세스, AI 데이터 엔지니어링 프로세스	데이터, 프레임워크	데이터의 명확한 이해와 활용을 지원하는 상세 정보를 제공하는가?	데이터, 메타데이터, 전처리 과정이 투명해야 함.
	데이터	데이터의 출처는 기록 및 관리되고 있는가?	데이터의 출처 혹은 추출 방식을 신뢰할 수 있어야 함.
	데이터	학습에 사용되는 특성을 분석하고 선정 기준을 마련하였는가?	선정된 데이터의 특성(feature)이 시험 검증 대상 구현에 적합한지 설명 가능해야 함.
	데이터	데이터 라벨링 시, 발생 가능한 편향을 확인하고 방지하였는가?	데이터의 레이블은 일관된 기준으로 작성되어야 함.
	데이터	데이터의 편향 방지를 위한 샘플링을 수행하였는가?	학습/검증/시험 데이터는 임의로 샘플링하여 데이터의 편향을 방지해야 함.
Design and Development Process: 구현 프로세스	프레임워크	오픈소스 라이브러리의 안정성을 확인하였는가?	신뢰할 수 있는 개발 프레임워크를 사용해야 함.
	프레임워크	오픈소스 라이브러리의 위험 요소는 관리되고 있는가?	개발 프레임워크의 라이선스 관계를 확인해야 하며, 알려진 보안 취약점을 관리해야 함.
	학습모델	사용자가 모델 추론 결과의 도출 과정을 수용할 수 있도록 근거를 제공하는가?	가급적이면 모델의 추론 결과를 설명할 수 있어야 함.
	학습모델	AI 모델 상세 문서를 통해 모델의 명세를 투명하게 제공하는가?	가급적이면 모델의 추론 결과를 설명할 수 있어야 함.
	학습모델	AI 모델 추론 결과에 대한 설명을 제공하는가?	가급적이면 모델의 추론 결과를 설명할 수 있어야 함.