ChatGPT

# AI 직무 평가: 주요 주제 요약

## 데이터 전처리 기법

- **데이터 수집:** AI 모델의 성능은 데이터 품질에 크게 의존하므로, 다양한 원천에서 **풍부하고 대표성 있는 데이터를 수집**하는 것이 중요합니다. 수집한 원본 데이터는 보통 오류와 잡음, 누락치로 **매우 지저분**하며 실제 분석에 바로 사용할 수 없기 때문에, 데이터 과학자의 업무 시간 중 약 80%가 **데이터 전처리**에 소요된다고 알려져 있습니다 [1].
- **데이터 정제:** 수집된 원시 데이터의 **오류를 수정**하고 일관성을 높이는 단계입니다. 누락된 값은 제거하거나 평균/중앙값 대치, 예측 모델 대치 등의 **결측치 처리** 기법으로 다룰 수 있습니다 [2] [3]. 또한 **이상치**(outlier)를 탐지하여 제거하거나 영향 완화하고, 데이터 간 **중복 레코드**를 식별해 제거함으로써 데이터의 정확도를 높입니다 [4] [5]. 이와 함께, 여러 출처에서 온 데이터를 통합하고 형식을 변환(예: 범주형 변수를 원-핫 인코딩)하거나 **정규화/스케일링**하여 변수 스케일을 맞추는 등 모델 학습에 적합한 형태로 데이터를 변환합니다 [6] [7]. 이러한 정제 과정을 통해 **데이터 품질 향상** 및 **쓰레기 입력 방지(Garbage in, garbage out)** 효과를 얻어 **모델 성능을 향상**시킬 수 있습니다 [8] [9].
- **데이터 증강: Data Augmentation**은 기존 데이터로부터 인위적으로 새로운 데이터를 생성하여 데이터셋의 크기와 다양성을 늘리는 기법입니다 [10]. 특히 **딥러닝** 모델에서는 대량의 훈련 데이터가 성능 향상에 중요하므로, 이미지나 텍스트 등의 분야에서 많이 활용됩니다. 예를 들어 이미지의 경우 **회전, 반전, 크기변환, 잡음 추가** 등의 변형을 통해 원본 이미지를 다양하게 바꿔 모델이 다양한 패턴을 학습하도록 합니다 [11]. 자연어 텍스트의 경우 **동의어 치환**이나 **역번역(back-translation)** 등을 통해 문장을 변형하여 데이터를 증강할 수 있습니다 [12] [13]. 이러한 증강 기법은 **오버피팅을 방지**하고 모델의 **일반화 성능을 개선**하는 데 효과적입니다.

## AI 모델 개발

- **아키텍처 설계:** 문제에 적합한 **AI 모델 구조를 설계**하는 것은 핵심 개발 과제입니다. 현대 딥러닝에서는 **합성곱 신경망(CNN)**, **순환 신경망(RNN)**, **트랜스포머(Transformer)** 등 **다양한 아키텍처 유형**이 존재하며, 데이터 형태와 문제 특성에 따라 적절한 구조를 선택하거나 새로 설계해야 합니다. 예를 들어 **트랜스포머**는 자기어텐션(self-attention) 메커니즘을 통해 **순차 데이터**를 병렬로 처리함으로써 전통적 RNN의 한계를 넘어 **NLP 작업의 효율과 성능을 혁신**하였고 [14], **ResNet**은 층 간 출력에 **직접 연결(skip connection)**을 추가하는 **잔차 학습**으로 **기존의 매우 딥(deep)한 신경망에서도 학습이 가능**하도록 만들어 컴퓨터 비전 분야의 정확도를 크게 끌어올렸습니다 [15]. 이러한 혁신 덕분에 수백 층 규모의 딥러닝 모델도 효과적으로 학습되었고, 이후 **DenseNet**, **EfficientNet**, **Vision Transformer** 등 보다 깊거나 효율적인 네트워크들이 등장하여 다양한 분야에 최적화된 아키텍처 설계가 이루어지고 있습니다. 아키텍처 설계 시에는 모델 용량(capacity)과 복잡도를 **데이터 양과 문제 난이도에 맞게 조절**하고, **과적합 방지**를 위한 규제 기법(Batch Norm, Dropout 등)도 함께 고려합니다. 최근에는 **신경망 아키텍처 검색(NAS)**이나 **AutoML** 기법을 통해 알고리즘이 자동으로 최적 아키텍처를 탐색하도록 하는 시도도 활발합니다.
- **설명 가능한 AI (XAI): 블랙박스**로 여겨지는 복잡한 AI 모델의 **의사결정 근거를 인간이 이해할 수 있도록 설명**하려는 기술 및 방법론입니다. 딥러닝 모델이 높은 예측 성능을 보여도, 내부 결정 로직이 불투명하면 실제 현업 적용에서 **신뢰성과 책임성** 문제가 생기므로, XAI 분야의 연구가 2018년 이후 활발해졌습니다 [16]. **국지적 기법**으로는 특정 예측에 가장 영향을 준 입력 특징을 추정하는 **LIME**이나 **SHAP** 같은 방법이 널리 쓰이고, **전역 기법**으로는 트리 모델의 **결정경로 시각화**나 신경망 가중치 규제 등을 통해 전체 구조의 해석력을 높이는 연구가 있습니다. 특히 딥러닝 비전 모델에서는 **Class Activation Map**이나 **Grad-CAM**처럼 최종 예측에 기여한 픽셀 영역을 **하이라이트**하여 보여주는 **시각적 설명 기법**이 개발되어 이미지 분류 결과를 설명하는 데 사용됩니다 [17]. 이외에도 **대시보드 형태의 설명 보고**(예: 특성 중요도 막대그래프, 부분의존도 그림)나 **대안 시나리오 제시**(counterfactual explanations) 등 다양한 XAI 접근법들이 존재합니다. XAI 기술을 활용하면 모델 사용자가

결과를 신뢰하고 모델의 오류 사례를 분석할 수 있으며, 금융이나 의료처럼 **설명 요구**가 높은 분야에서 **규제 준수**를 만족하면서 AI를 활용할 수 있게 됩니다 [18] [16].

- **모델 학습 및 평가: 모델 학습** 단계에서는 주어진 훈련 데이터에 대해 최적화 알고리즘(예: SGD, Adam 등)을 사용해 모델 파라미터를 갱신하며, 동시에 **검증 데이터**를 통해 일반화 성능을 점검합니다. 이때 **학습曲선(Learning Curve)**을 모니터링하여 **과적합**(overfitting)이나 **과소적합**(underfitting) 여부를 진단하는 것이 중요합니다 [19] [20]. 예를 들어 **과적합**이라면 **훈련 손실**은 계속 감소하여 거의 0에 가깝게 낮아지지만, **검증 손실**은 어느 시점 이후 더 이상 감소하지 않다가 오히려 증가세로 돌아서게 됩니다 [21]. 이때 훈련/검증 손실 곡선 사이에 **큰 격차**가 나타나며, 이는 모델이 학습 데이터의 패턴뿐 아니라 **노이즈까지 암기**하여 새로운 데이터에 성능이 떨어지는 전형적인 징후입니다 [22] [23]. 반대로 **과소적합** 상태에서는 **훈련 손실** 자체가 높게 남아 있고 검증 손실도 비슷하게 높은 수준을 유지하여, 두 곡선 사이 **격차는 크지 않지만 전반적으로 오류율이 높게** 나타납니다 [24]. 이 경우 모델 복잡도를 높이거나 더 많은 특징을 학습하도록 개선해야 합니다. 모델 평가 단계에서는 **평가 지표**를 사용해 성능을 정량적으로 측정합니다. 분류 문제에서는 **정확도(Accuracy)** 외에도 **정밀도(Precision)**, **재현율(Recall)**, **F1-점수** 등의 지표를 활용합니다. 특히 데이터 **클래스 불균형**이 심한 경우 단순 정확도는 착시를 줄 수 있어(예: 99%가 정상이고 1%가 이상치인 데이터에서 무조건 정상으로 예측하면 99% 정확도) [25], 실제 긍정 예측 중 얼마나 실제 긍정인가를 보는 **정밀도**, 실제 긍정 샘플 중 놓치지 않고 잡아낸 비율인 **재현율**, 두 값의 조화평균인 **F1** 등의 지표가 더욱 **유용한 통찰**을 제공합니다 [26] [25]. 예컨대 **스팸 필터링**에서는 중요한 메일이 스팸으로 잘못 차단되지 않도록 **정밀도**가 중요하고, **암 환자 진단** 모델에서는 환자를 놓치지 않는 **재현율**이 중요하듯이, 문제 성격에 따라 적절한 평가 척도를 선택해야 합니다 [27] [28]. 회귀 문제에서는 **평균제곱오차(MSE)**, **평균절대오차(MAE)**, **$R^2$ 점수** 등이 활용되며, 랭킹/검색 문제에서는 **AUC**(곡선하면적)나 **MAP** 등이 쓰입니다. 다양한 지표를 종합적으로 고려해 모델을 평가함으로써 모델의 강점과 약점을 파악하고 향후 개선 방향을 도출할 수 있습니다.
- **모델 튜닝 기법:** 최적의 모델 성능을 얻기 위해 **하이퍼파라미터 최적화(Hyperparameter Optimization, HPO)**와 **불균형 데이터 처리** 등의 튜닝 기법을 적용합니다.
- 하이퍼파라미터 최적화: 학습률, 은닉층 크기, 정규화 계수 등 모델의 하이퍼파라미터를 체계적으로 탐색하여 최적 조합을 찾는 과정입니다. **그리드 탐색(Grid Search)**과 **랜덤 탐색(Random Search)**이 전통적으로 가장 많이 사용되는 접근법이며, 전자는 미리 정한 모든 조합을 전수 평가하고 후자는 설정한 횟수만큼 무작위 조합을 평가합니다 [29]. 다만 그리드/랜덤 탐색은 **탐색 범위**에 포함된 값들만 평가하고 **연속 변수도 이산 샘플링**하므로 비효율적일 수 있습니다 [30] [31]. 최근에는 **베이지안 최적화**가 **보다 적은 시도로 우수한 하이퍼파라미터를 찾는** 강력한 방법으로 각광받고 있습니다 [32]. 베이지안 방법은 이전 시도들의 평가 결과를 바탕으로 다음 실험 지점을 **똑똑하게 선택**하여, 불필요한 영역을 배제하고 **더 빠르게 수렴**합니다 [33] [34]. 이외에도 **진화 알고리즘**, **Hyperband** 등의 기법과 Optuna, Ray Tune 같은 HPO 프레임워크를 활용하여 효율적으로 모델 튜닝을 수행합니다.
- 클래스 불균형 해결: 현실 데이터셋에서는 특정 클래스가 매우 드문 경우가 많아(Model의 편향 초래) 이를 보정해줘야 합니다. **데이터 수준 방법**으로는 **오버샘플링**과 **언더샘플링**이 기본적입니다. 오버샘플링은 **소수 클래스 데이터를 복제**하거나 **합성하여(SMOTE 등)** 데이터 비율을 높이는 방법이고 [35], 언더샘플링은 다수 클래스 데이터 중 일부를 제거하여 균형을 맞추는 기법입니다 [36]. 예를 들어 `imblearn` 라이브러리의 `RandomOverSampler`나 `SMOTE`를 이용해 소수 클래스 샘플을 증가시킬 수 있습니다 [37] [38]. **알고리즘 수준 방법**으로는 **비용 민감 학습**이 있습니다. 이는 모델 학습 시 **클래스별 가중치**를 달리 부여해, 희소한 클래스의 오류에 더 큰 페널티를 주는 방식입니다 [39]. 사이킷런의 `class_weight='balanced'` 옵션이나 XGBoost의 `scale_pos_weight` 파라미터 등이 이러한 역할을 합니다 [40] [41]. 더 나아가 **Focal Loss**처럼 **어려운 예제에 가중치를 더 부여**하는 특수 손실함수도 많이 쓰입니다 [42]. Focal Loss는 분류기가 쉽게 맞히는 다수 클래스 샘플에서는 손실 기여도를 낮추고, 소수 클래스처럼 오분류하기 쉬운 샘플에 더 큰 가중 손실을 부여함으로써 **불균형 데이터를 효과적으로 학습**하도록 설계된 기법입니다 [42]. 이러한 튜닝 기법을 통해 **데이터 불균형으로 인한 성능 저하**나 **하이퍼파라미터 부적절 설정으로 인한 미최적화**를 방지하고, 모델의 성능을 극대화할 수 있습니다.

## AI 시스템 구축

- **ML 파이프라인 설계 및 배포:** 데이터 수집부터 모델 배포까지의 과정을 **자동화된 파이프라인**으로 구성하는 것이 중요합니다. 일반적인 ML 파이프라인은 데이터 수집/전처리 → 모델 학습 → 모델 검증 → 배포의 단계를 거

치며, 이를 효율적으로 운영하기 위해 **MLOps** 개념이 등장했습니다. MLOps는 소프트웨어 개발의 **DevOps** 원칙을 머신러닝에 접목한 것으로, **CI/CD(지속적 통합/전달)** 뿐 아니라 **CT(지속적 학습)**를 통해 모델을 지속적으로 개선/배포하는 전략입니다 [43] [44] . **Google** 등에 따르면 "MLOps 실천이란 통합, 테스트, 릴리즈, 배포, 인프라 관리 등 ML 시스템 구축의 모든 단계에 자동화와 모니터링을 적용하는 것"이라고 정의됩니다 [45] . 이를 구현하기 위해 **파이프라인 오케스트레이션 도구**(Kubeflow, Airflow 등)를 사용해 **데이터 준비, 학습, 평가, 배포 단계**를 연결하고, **컨테이너화** 및 **인프라스 코드(IaC)**로 일관된 환경에서 재현성을 확보합니다. 또한 모델을 **REST API나 Microservice**로 패키징하여 클라우드나 엣지 서버에 배포하고, AB 테스트나 점진적 롤아웃을 통해 안정적으로 사용자 트래픽에 노출합니다 [46] [47] . **Feature Store**를 구축해 온라인/오프라인 특성 일관성을 관리하거나, **모델 버전관리**를 통해 이전 모델과 성능을 비교하며, **데이터 및 모델에 대한 형상관리(DVC, Model Registry)**도 적용합니다. 이러한 체계적인 파이프라인 설계는 모델 개발부터 서비스 배포까지의 사이클을 단축하고, **지속적인 업데이트**를 가능케 하여 비즈니스 가치를 신속히 제공합니다 [44] [48] .

- **AI 시스템 모니터링 및 자동화:** 모델이 운영 환경에 배포된 이후에는 **지속적인 모니터링**을 통해 성능 저하나 데이터 이상을 감지하고 자동으로 대응하는 것이 중요합니다. **모델 모니터링** 측면에서, **예측 성능 지표**(정확도, 응답시간 등)를 **실시간으로 추적**하여 임계치 미달 시 경고를 발생시키거나 **재학습 트리거**를 거는 체계를 갖춥니다. 특히 입력 데이터 분포나 관계가 훈련 시와 달라지는 **드리프트(drift)**를 탐지하는 것이 핵심입니다. **데이터 드리프트**는 입력 특성들의 분포 변화(예: 이미지 밝기나 텍스트 어휘의 변화)이고, **개념 드리프트**는 입력과 출력 간의 관계 변화(예: 소비자 행동 패턴 변화로 모델이 학습한 관계가 무효화됨)를 의미합니다 [49] . **개념 드리프트** 발생 시에는 기존 모델이 더 이상 올바른 예측을 못하게 되므로 **훈련 데이터 갱신 및 모델 재학습**이 필요합니다 [49] . 한편 **데이터 드리프트**는 개념 드리프트의 **전조 현상**으로 볼 수 있어, 입력 데이터 분포가 유의미하게 달라지는 조짐을 조기에 포착하면 본격적인 성능 저하 전에 **선제적으로 대응(모델 재훈련 등)**할 수 있습니다 [50] . 예를 들어 **모니터링 도구**를 통해 모델 입력 데이터의 통계량(KL 다이버전스 등으로 측정)을 지속 추적하고, 일정 임계치를 넘으면 자동으로 **새 데이터로 모델을 재훈련**하는 파이프라인을 가동합니다 [50] . 이외에도 시스템 모니터링 측면에서 **서버 리소스 사용률(GPU/메모리)**, **처리량(QPS)**, **에러 율** 등을 추적하여 **운영 안정성**을 유지합니다. **자동화** 측면에서는 정기적으로 파이프라인이 동작하도록 **스케줄러**를 설정하고, 새로운 데이터 발생 시 **이벤트 트리거 방식**으로 실시간 학습/배포가 일어나게 할 수 있습니다 [51] [52] . 나아가 **모델 성능 저하 감지 → 데이터 큐레이션 → 재학습 → 배포**까지 완전히 자동화된 **피드백 루프**를 구축하면, 사람이 개입하지 않아도 AI 시스템이 **자체적으로 진화**하고 문제를 수정해나갈 수 있습니다 [53] [54] . 이러한 모니터링 및 자동화는 **모델의 신뢰성**과 **서비스 수준 목표(SLO)**를 유지하는 데 필수적입니다.
- **AI 시스템 최적화:** AI 모델을 실제 서비스에 활용하려면 **응답 지연 최소화**, **처리량 향상**, **인프라 비용 절감** 등을 위한 최적화가 요구됩니다. **모델 경량화**는 대표적인 최적화 기법으로, 불필요한 복잡도를 줄여 **추론 속도**를 높이고 **메모리 사용량**을 줄입니다. 예를 들어 **모델 가지치기(pruning)**는 중요도가 낮은 뉴런이나 가중치를 제거하여 모델 크기를 줄이고 계산량을 감소시킵니다. **양자화(quantization)** 기법은 모델 가중치와 연산을 32비트 부동소수 대신 16비트나 8비트 정밀도로 표현하여 **메모리 사용량과 연산량을 크게 줄이는** 방법입니다 [55] . 또한 **지식 증류(Knowledge Distillation)**는 복잡한 **대형 모델(교사)**의 지식을 **경량 모델(학생)**에 옮겨서, 성능은 유지하면서도 훨씬 **작고 빠른 모델**을 얻는 최적화 전략입니다 [55] . 이러한 모델 수준 최적화 외에도, **병렬 분산 처리**와 **하드웨어 가속**을 통해 시스템을 최적화합니다. 예를 들어 **멀티스레딩**이나 **멀티-GPU 분산 추론**으로 일처리량을 높이고, CUDA/TensorRT와 같은 **최적화 라이브러리**를 활용하여 하드웨어의 최대 성능을 끌어냅니다. **배치 처리 크기**를 조절하거나 **동적 연산 그래프 최적화**(런타임 최적화)로 지연 시간을 단축할 수도 있습니다. 이밖에 **캐싱**을 통해 동일한 입력에 대한 반복 연산을 줄이고, **메모리 최적화 기법**으로 CPU-GPU 간 전송 병목을 줄이는 등 시스템 전반에 걸친 튜닝을 실시합니다. 마지막으로 이러한 최적화의 효과와 트레이드오프(예: 양자화로 인한 미세 성능 감소)를 **평가 지표**로 모니터링하면서 최적의 균형점을 찾아 적용합니다 [56] [57] . 결과적으로 AI 시스템 최적화를 통해 **실시간 서비스**에서 요구하는 성능을 만족하고, **비용 효율적**으로 모델을 운영할 수 있게 됩니다.

## 주요 AI 기술 트렌드 (2018년~현재)

- **트랜스포머와 초거대 언어모델:** 2017년 말 등장한 **트랜스포머(Transformer)** 아키텍처는 이후 NLP 분야를 지배하는 표준이 되었습니다. 트랜스포머는 **자기어텐션 메커니즘**을 통해 RNN 없이도 문장 내 장거리 의존성을 효과적으로 포착하며 병렬 연산이 가능해, **번역, 질의응답, 요약** 등 NLP 성능을 혁신적으로 향상시켰습니다 [14] . 2018년 등장한 **BERT**는 트랜스포머를 **양방향(bidirectional)**으로 학습하여 문맥 이해 능력을 끌어올렸고, 2020년 발표된 **GPT-3**는 **1750억 개**에 이르는 파라미터를 가진 초대규모 **사전학습 언어모델**로서 인간에 가

까운 자연어 생성 능력을 보여주었습니다 [58]. 이러한 **거대 언어모델(LLM)**의 계보는 곧 **GPT-4** 등으로 이어졌고, 2022년 말 공개된 OpenAI의 **ChatGPT**는 일반 대중에게도 **생성형 AI**의 강력함을 인식시킨 사건으로 평가됩니다 [59]. 현재 LLM들은 **소규모 감독** 또는 **미세조정(fine-tuning)**만으로도 새로운 NLP 태스크에 높은 성능을 보이며, 지식 검색, 코딩 보조, 대화형 에이전트 등 **산업 전반에 변혁**을 일으키고 있습니다. 또한 트랜스포머 구조는 **비전**(Vision Transformer, 2020)과 **음성** 등 다른 도메인에도 응용되어 **범용 모델 아키텍처**로 자리잡고 있습니다 [60].

- **생성형 AI의 부상:** 2014년 **GAN**(생성적 적대 신경망)의 발명 이후로 AI가 새로운 데이터를 창조해내는 **생성형 모델** 연구가 급성장했습니다. 이미지 생성, 동영상 생성, 텍스트 생성 등 다양한 분야에서 생성 모델들이 등장했으며, 2018~2019년에는 **GAN** 기반으로 놀랍도록 현실적인 **딥페이크** 영상과 사진이 화제가 되기도 했습니다. 2022년에는 OpenAI의 **DALL-E 2**가 문자 설명만으로도 고해상도 이미지를 만들어내어 전세계적인 관심을 모았고, 뒤이어 공개된 **Stable Diffusion** 등 **확산 모델(Diffusion Model)** 기반 생성기도 예술, 디자인 분야에 큰 반향을 일으켰습니다 [61] [62]. 한편 텍스트 생성에서는 앞서 언급한 GPT 계열 모델이 주도하여, **블로그 글**, **시나리오**, **프로그램 코드**까지 자동 생성하는 **자연어 생성** 활용이 폭발적으로 늘었습니다 [63]. 이러한 생성형 AI 기술은 **콘텐츠 생산성**을 높이고 새로운 **비즈니스 기회**를 창출하지만, 한편으로 **허위정보 생성**이나 **저작권** 이슈 등의 윤리적 문제도 대두되어 이에 대한 대응 연구도 활발합니다. 그럼에도 생성형 AI는 **2020년대 핵심 기술 트렌드**로 자리매김했으며, 이미지-텍스트 간 상호생성, 멀티모달 생성(텍스트→영상, 텍스트→3D 모델 등)처럼 다양한 형태로 진화하고 있습니다 [64] [65].
- **컴퓨터 비전 모델 혁신: 이미지 인식**을 비롯한 컴퓨터 비전 분야에서는 2012년 **AlexNet**을 시작으로 한 **CNN 아키텍처** 발전이 이어져왔고, 2015년 **ResNet**의 등장으로 **초딥 신경망**의 효과적인 학습이 가능해진 것이 큰 전환점이었습니다 [15]. ResNet은 152층까지 누적된 매우 깊은 네트워크로 **ImageNet 대회**를 제패하며 사실상 CV 모델의 기본 구성이 되었고, 이후 **ResNeXt**, **WideResNet**, **DenseNet** 등 변종들이 나와 특성 추출 능력을 높였습니다. 2017년에는 **셀프-어텐션** 구조를 결합한 **Non-local Neural Network**가 비전에도 주목받았고, 2019년 구글의 **EfficientNet**은 **네트워크 깊이·너비·해상도를 균형 있게 스케일업**하는 **Compound Scaling**으로 **적은 파라미터로도 높은 정확도**를 달성하여 실용적인 모델 설계 방향을 제시했습니다 [66] [67]. 2020년에는 NLP 혁신이었던 **Transformer**를 이미지 처리에 적용한 **Vision Transformer(ViT)**가 제안되어, 대규모 데이터에서 **Conv 레이어 없이도 SOTA 수준 시각 인식**이 가능함을 보였습니다 [60]. 이후 ViT를 개선한 **SWIN Transformer** 등 **어텐션 기반 비전 모델**들이 등장하며 **CNN과 어텐션의 융합**이 이루어지고 있습니다. 또한 **자기지도 학습**과 **대량의 사전학습(backbone)** 트렌드에 따라, **CLIP** 같은 거대 멀티모달 모델을 비전 분야에 활용하고 후속 작업에 파인튜닝하는 **Transfer Learning**이 표준화되었습니다. 요약하면, 컴퓨터 비전에서도 **모델 아키텍처의 지속적 발전**과 **대규모 데이터/모델 활용**이라는 두 축이 2018년 이후 현재까지 주요 트렌드입니다.
- **연합학습(Federated Learning)과 프라이버시 강화:** 데이터 **프라이버시**에 대한 관심이 높아지면서, 2017년경부터 제안된 **연합학습**이 실제 적용 단계로 접어들었습니다. 연합학습은 **훈련 데이터를 중앙 서버에 모으지 않고**도 각 기기에서 **지역(local) 모델을 학습**하고 이를 서버에서 합치는 방식으로 **분산된 환경에서 공동 학습**을 가능케 합니다 [68]. 예를 들어 스마트폰 키보드의 단어 추천 모델을 위해 사용자의 입력 데이터를 서버로 보내지 않고 기기 내에서 학습시킨 뒤 모델 업데이트만 공유하는 식입니다. 이 방법은 개인정보 노출 위험을 줄이고, **데이터가 분산된 상태에서도 글로벌 모델**을 구축할 수 있다는 장점이 있어 **모바일 AI 및 의료AI** 분야에서 주목받고 있습니다 [68]. 구글은 **TensorFlow Federated** 프레임워크를 공개했고, 다양한 기업에서 이를 활용한 **프라이버시 보존 ML**을 연구 중입니다 [69]. 한편 **엣지 AI** 역시 트렌드로, 센서가 달린 엣지 디바이스(카메라, IoT장치 등) 자체에서 **경량화된 ML 추론**을 수행하는 사례가 늘고 있습니다. Gartner 보고서에 따르면 **AIoT**라 불리는 AI+IoT 융합 영역에서, 2025년까지 기업 IoT 프로젝트의 80% 이상에 AI 기능이 포함될 것으로 전망됩니다 [70]. 이는 곧 네트워크 지연이나 프라이버시 이슈 없이 **현장에서 실시간 의사결정**을 내리는 **분산 AI 시스템**의 시대가 오고 있음을 의미합니다. 이 밖에도 **강화학습(RL)의 발전**(알파스타, OpenAI Five 등), **그래프 신경망(GNN)의 부상**(소셜 네트워크 및 추천시스템 활용) 등도 2018년 이후 두드러진 흐름입니다. 전반적으로 **대규모 사전학습 모델의 활용**, **멀티모달 통합**, **자동화된 ML파이프라인(MLOps)**, **프라이버시 및 윤리적 AI**가 현재 AI 분야의 화두이며, 기업들은 이러한 기술 트렌드를 빠르게 업무에 도입해 **서비스 혁신**과 **경쟁력 강화**를 도모하고 있습니다 [71] [72].

---

1 4 5 8 9 Data Preprocessing in Machine Learning: Steps & Best Practices
https://lakefs.io/blog/data-preprocessing-in-machine-learning/

2 3 6 7 10 11 12 13 Data Preprocessing: A Complete Guide with Python Examples | DataCamp
https://www.datacamp.com/blog/data-preprocessing

14 58 60 A Comprehensive Review of Deep Learning: Architectures, Recent Advances, and Applications
https://www.mdpi.com/2078-2489/15/12/755

15 Understanding ResNet: A Deep Dive into Residual Neural Networks | by Rohan Mistry | Medium
https://medium.com/@rohanmistry231/understanding-resnet-a-deep-dive-into-residual-neural-networks-6d8c8c227fd0

16 17 18 Explainable AI (XAI): A survey of recents methods, applications and frameworks | AI Summer
https://theaisummer.com/xai/

19 Learning Curve to identify Overfitting and Underfitting in Machine ...
https://medium.com/data-science/learning-curve-to-identify-overfitting-underfitting-problems-133177f38df5

20 Understanding Overfitting, Underfitting, and Learning Curves in ...
https://medium.com/@aarishalam22/understanding-overfitting-underfitting-and-learning-curves-in-machine-learning-fe19825125c8

21 22 23 24 Learning Curve To Identify Overfit & Underfit - GeeksforGeeks
https://www.geeksforgeeks.org/machine-learning/learning-curve-to-identify-overfit-underfit/

25 26 27 28 Understanding Precision, Recall, and F1 Score Metrics | by Piyush Kashyap | Medium
https://medium.com/@piyushkashyap045/understanding-precision-recall-and-f1-score-metrics-ea219b908093

29 30 31 32 33 34 Grid Search vs. Random Search vs. Bayesian Optimization
https://blog.dailydoseofds.com/p/grid-search-vs-random-search-vs-bayesian

35 36 37 38 39 40 41 42 Advanced Class Imbalance Handling: From Basics to Super-Advanced | by Adnan Mazraeh | Medium
https://medium.com/@adnan.mazraeh1993/advanced-class-imbalance-handling-from-basics-to-super-advanced-65722f59c21b

43 44 45 48 54 MLOps: Continuous delivery and automation pipelines in machine learning | Cloud Architecture Center | Google Cloud
https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

46 47 51 52 53 MLOps Architecture Guide
https://neptune.ai/blog/mlops-architecture-guide

49 50 Machine learning model monitoring: Best practices | Datadog
https://www.datadoghq.com/blog/ml-model-monitoring-in-production-best-practices/

55 56 57 Deep Learning Model Optimization Methods
https://neptune.ai/blog/deep-learning-model-optimization-methods

59 61 62 63 64 65 68 69 70 72 2023년 기대되는 AI 기술 트렌드 10가지 - 디지털 인사인트 매거진
https://digit2sight.com/2023%eb%85%84-%ea%b8%b0%eb%8c%80%eb%90%98%eb%8a%94-ai-%ea%b8%b0%ec%88%a0-%ed%8a%b8%eb%a0%8c%eb%93%9c-10%ea%b0%80%ec%a7%80/

66 67 EfficientNet: Optimizing Deep Learning Efficiency
https://viso.ai/deep-learning/efficientnet/

71 2023년 기대되는 AI 기술 트렌드 10가지 | 커리어리
https://careerly.co.kr/comments/75278