
# (7주차) 거대언어모델(Large Language Model)

명지대학교
MYONGJI UNIVERSITY

## 거대언어모델(LLM) 개요

### LLM 개요

**LLM(Large Languate Model) 이란?**

방대한 양의 데이터를 기반으로 하고 다수의 파라미터를 학습시킨
사전 훈련된 매우 거대한 딥러닝 언어모델

자연어 및 기타 콘텐츠 유형을 이해하고 생성하여
광범위한 작업을 수행할 수 있는 능력을 갖춘 언어모델

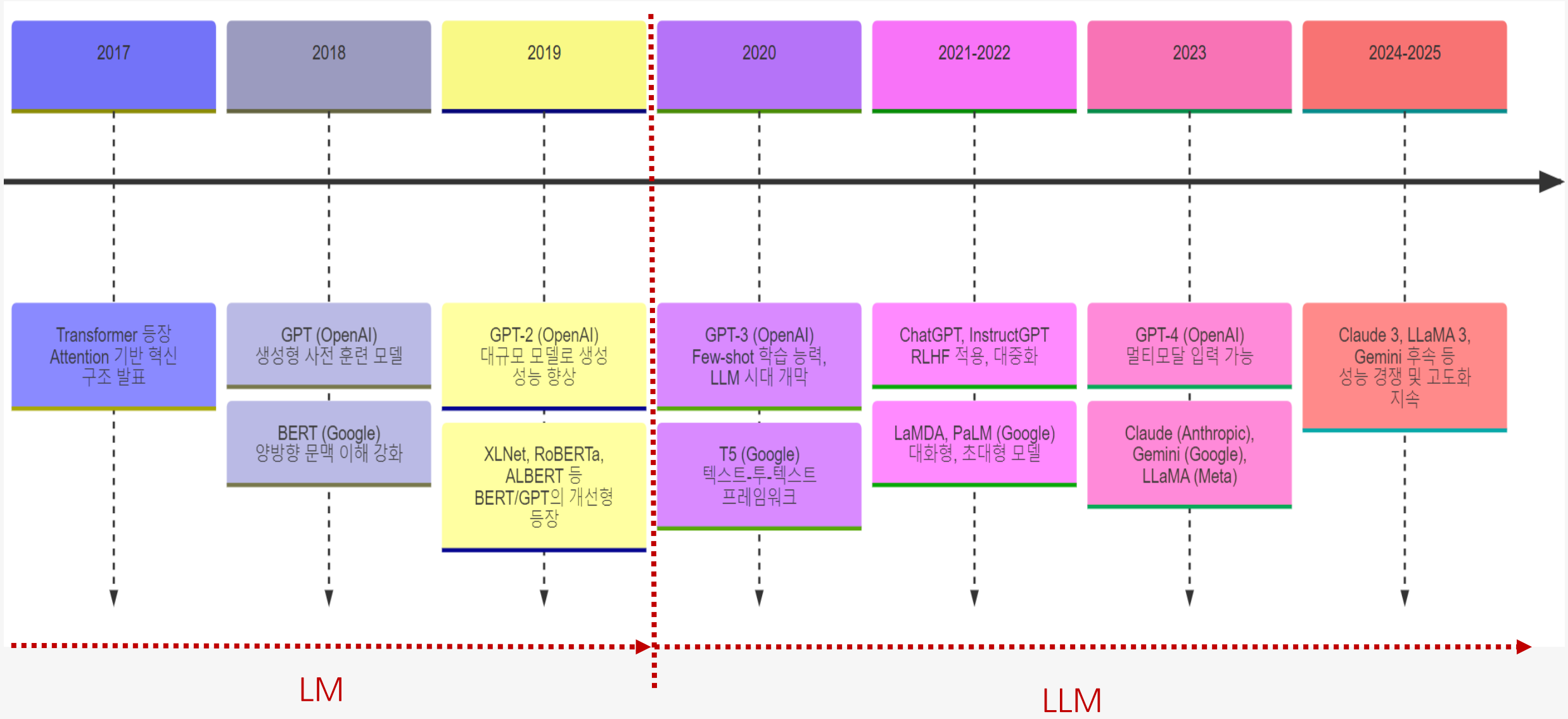
# 거대언어모델(LLM) 개요

## LLM / sLLM / SLM 비교

| 구분 | LLM (Large Language Model) | sLLM / SLM (Small Language Model) |
|---|---|---|
| 명칭 | 대규모 언어 모델 | 소형 언어 모델 (sLLM은 종종 SLM과 혼용되거나 약간 더 큰 SLM을 지칭) |
| 파라미터 수 (규모) | 수백억 개 ~ 수조 개 이상 | 수백만 개 ~ 수십억 개 수준 |
| 학습 데이터 규모 | 매우 방대함 (인터넷 규모의 텍스트 등) | 상대적으로 작거나 특정 도메인/작업에 특화된 데이터 |
| 컴퓨팅 자원 (학습/추론 비용) | 매우 높음 (고성능 GPU 클러스터 필요) | 낮음 (단일 GPU 또는 CPU에서도 실행 가능) |
| 성능 및 능력 | • 범용적이고 뛰어난 성능<br>• 복잡한 추론, 창의적 생성 능력<br>• 광범위한 지식 보유 | • 특정 작업이나 도메인에 최적화<br>• LLM 대비 일반 능력은 제한적일 수 있음<br>• 가볍고 빠름 |
| 응답 속도 (지연 시간) | 상대적으로 느릴 수 있음 | 빠름 |
| 비용 (개발/운영) | 높음 | 낮음 |
| 주요 응용 분야 | • 범용 챗봇 (ChatGPT, Gemini 등)<br>• 복잡한 문제 해결<br>• 콘텐츠 생성<br>• 코드 생성 | • 엣지 디바이스 (스마트폰, IoT 기기)<br>• 특정 업무 자동화 (고객 응대, 문서 요약)<br>• 빠른 응답이 필요한 서비스<br>• 온프레미스 환경 |
| 주요 특징 및 목표 | • 최대한의 성능과 범용성 추구<br>• 최첨단 AI 능력 구현 | • 효율성, 경제성, 속도 중시<br>• 특정 환경 및 작업 최적화<br>• 접근성 및 사용 편의성 |

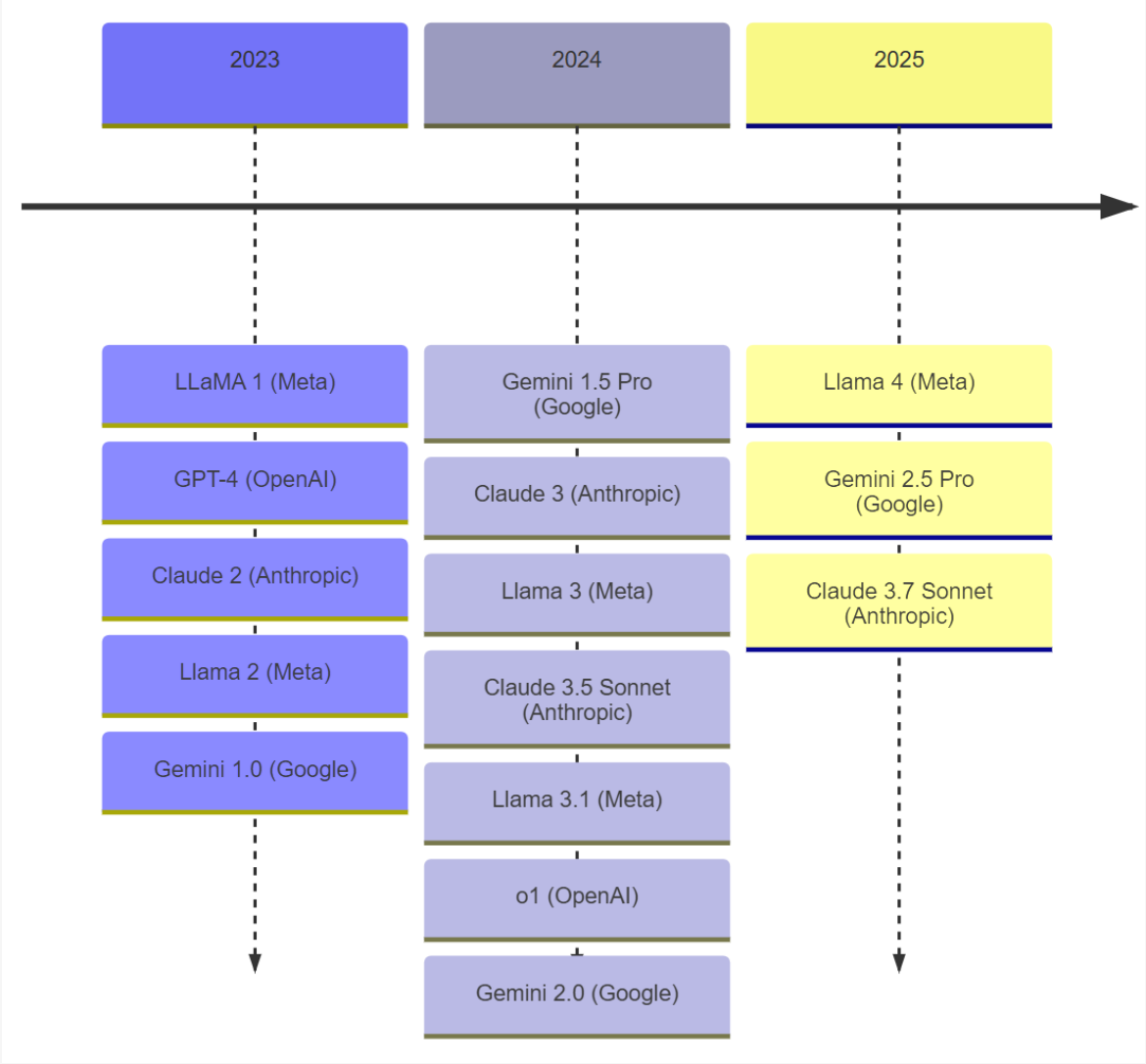
# 거대언어모델(LLM) 개요

## 거대언어모델의 발전과정



# 거대언어모델(LLM) 개요

## 최근 등장 거대언어모델



# 거대언어모델(LLM) 개요

## 최근 등장 거대언어모델

| 연도 | 주요 모델 (개발사) | 토큰 수 (컨텍스트 창) | 주요 특징/기여 |
|---|---|---|---|
| 2023 | LLaMA 1 (Meta) | 2K | 고성능 오픈소스 모델 (7B-65B) |
| 2023 | GPT-4 (OpenAI) | 8K / 32K (Turbo: 128K) | 멀티모달(텍스트/이미지 입력), 향상된 추론 |
| 2023 | Claude 2 (Anthropic) | 100K | 안전성 강조, 100K 컨텍스트 창 |
| 2023 | Llama 2 (Meta) | 4K | Llama 1 개선, 상업적 이용 가능한 오픈 모델 |
| 2023 | Gemini 1.0 (Google) | Pro: 32K, Ultra: 1M (추정) | 네이티브 멀티모달 모델 (Ultra, Pro, Nano) |
| 2024 | Gemini 1.5 Pro (Google) | 1M | 1M 토큰 컨텍스트 창 |
| 2024 | Claude 3 (Anthropic) | 200K | Opus, Sonnet, Haiku 모델군, 멀티모달 |
| 2024 | Llama 3 (Meta) | 8K | Llama 2 개선 (8B, 70B) |
| 2024 | Claude 3.5 Sonnet (Anthropic) | 200K | Claude 3 Sonnet 성능 개선 |
| 2024 | Llama 3.1 (Meta) | 128K | 405B 파라미터 모델 추가 |
| 2024 | o1 (OpenAI) | 128K (추정) | 추론(Reasoning) 특화 모델 |
| 2024 | Gemini 2.0 (Google) | 1M (Flash-Lite 기준) | Flash, Flash-Lite 모델, 실시간 API |
| 2025 | Llama 4 (Meta) | Scout: 10M | 네이티브 멀티모달, MoE, 장문맥 (Scout, Maverick, Behemoth) |
| 2025 | Gemini 2.5 Pro (Google) | 1M | 1M 컨텍스트, 향상된 추론/멀티모달 |
| 2025 | Claude 3.7 Sonnet (Anthropic) | 200K | 하이브리드 추론, 확장된 사고 |

## GPT3 언어모델

### GPT3

- 웹 페이지 크롤링을 통해 수집한 4990억개 데이터셋 중에서 가중치 샘플링을 통해서 3000억(300B)개로 구성된 데이터셋으로 사전 학습(pre-training)
- 모델 파라미터 수는 1750억개

Fine-tuning 없이
자연어 처리 벤치마크 테스트에서 최고의 성능 달성

OpenAI는 훈련된 모델 또는 전체 소스코드는 미공개

## ChatGPT

### ChatGPT 개요

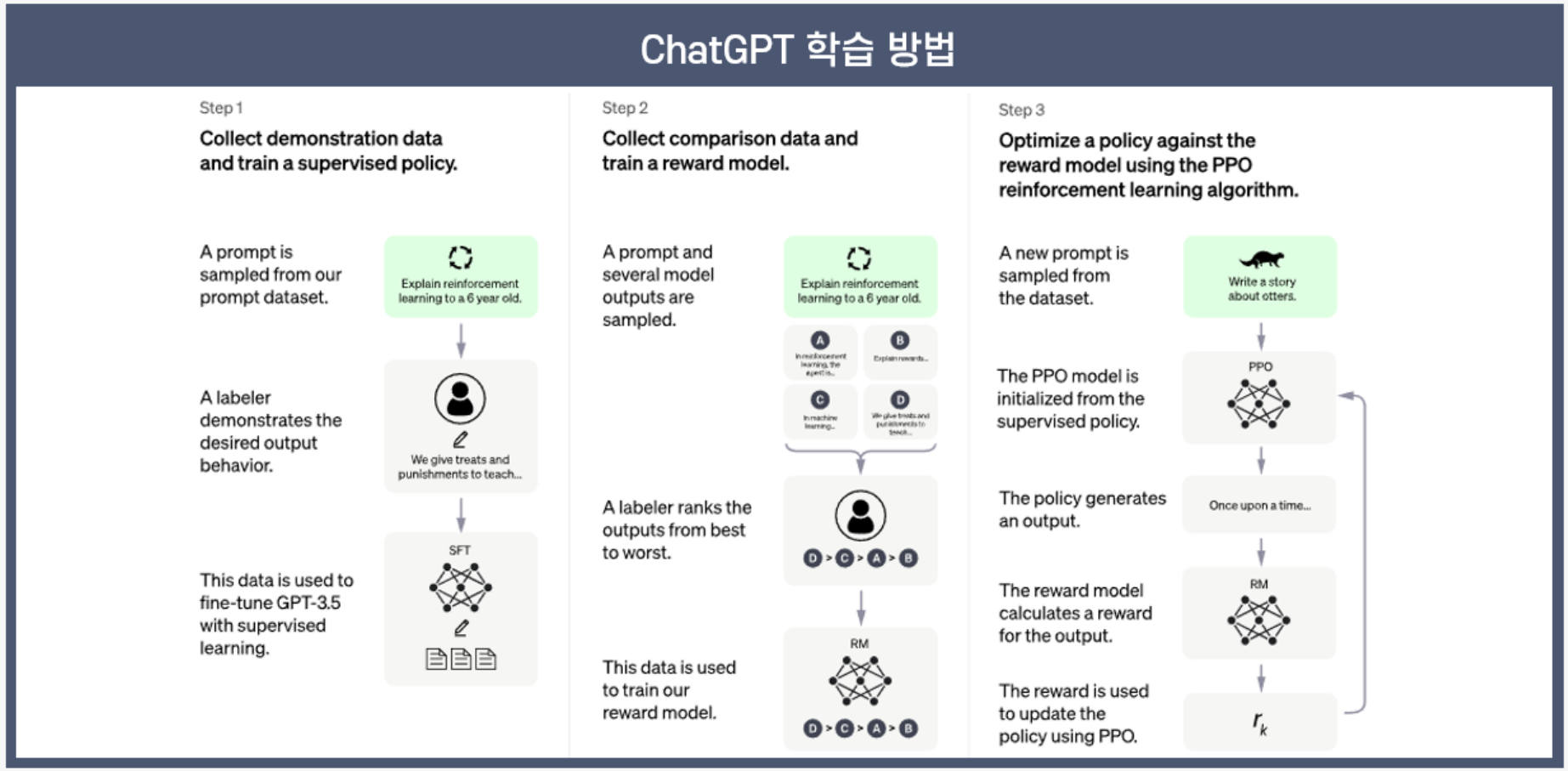
**ChatGPT(Chat + Generative Pre-trained Transformer)란?**

2021년 12월 OpenAI가 개발한 인공지능 대화 시스템

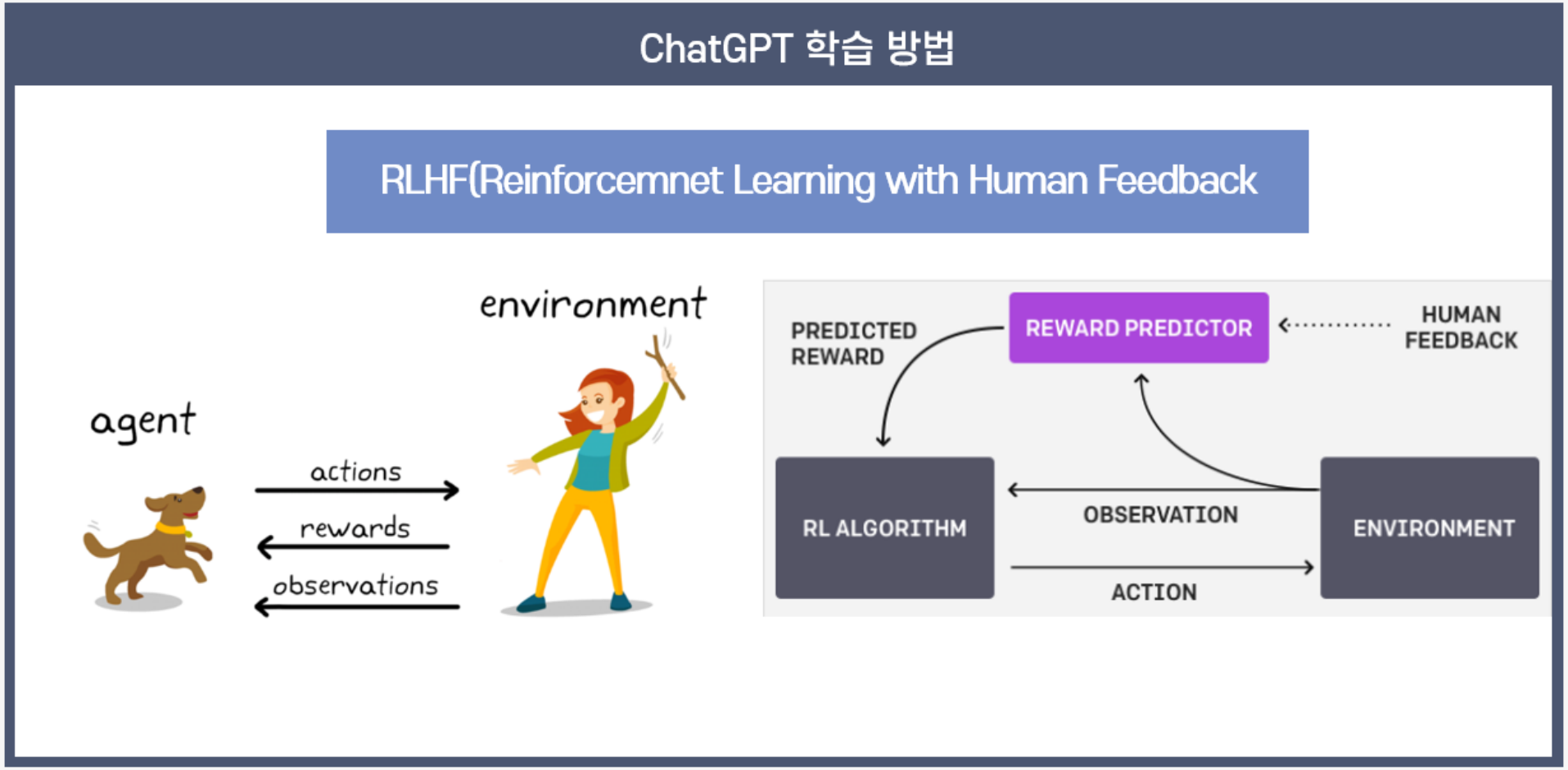
지도학습(Supervise Learning)과 강화학습(Reinforcement Learning) 활용
GPT3.5 언어모델 미세조정(fine-tuning)

# 거대언어모델(LLM)

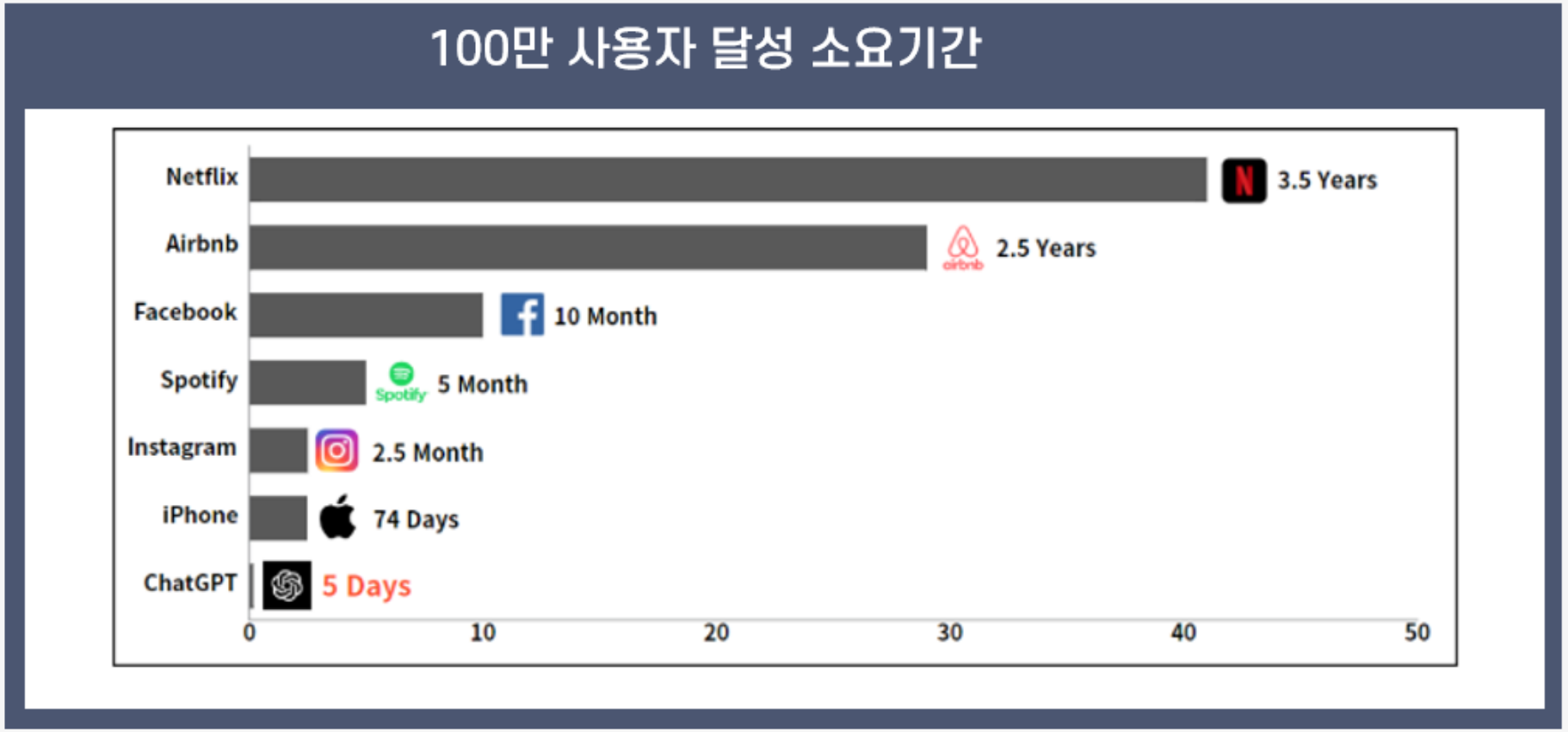
- ChatGPT
  - ChatGPT 학습방법



- ChatGPT
  - ChatGPT 학습방법



- ChatGPT
  - ChatGPT 등장과 패러다임의 변화



## 최신 언어모델 비교 분석

### GPT-4o & GPT-4o mini (OpenAI)

**GPT-4o**

- 특징: OpenAI 최초 네이티브 멀티모달 (텍스트, 오디오, 이미지 입출력)
- 실시간 오디오 상호작용 (평균 320ms 응답 속도)
- GPT-4 Turbo 대비 API 비용 50% 저렴
- 비영어권 언어 처리 성능 개선
- 128K 토큰 컨텍스트 창 , 2023년 10월까지 지식 학습
- 성능: 영어/코드(GPT-4 Turbo 동등), 비영어/비전/오디오(향상)
- 고유 기능: 단일 종단간 모델로 처리 속도/상호작용성 극대화 , 자연스러운 음성 AI 가능성

## 최신 언어모델 비교 분석

### GPT-4o & GPT-4o mini (OpenAI)

| | |
|---|---|
| GPT-4o mini | - 특징: GPT-4o 소형 버전, 비용 효율성 극대화<br>- GPT-3.5 Turbo 보다 저렴, 더 높은 벤치마크 성능<br>- 지시 계층(Instruction Hierarchy) 보안 적용<br>- 성능: MMLU, MGSM, HumanEval 등에서 경쟁 소형 모델 능가<br>- 고유 기능: 성능-비용 균형으로 AI 접근성 향상 |

## 최신 언어모델 비교 분석

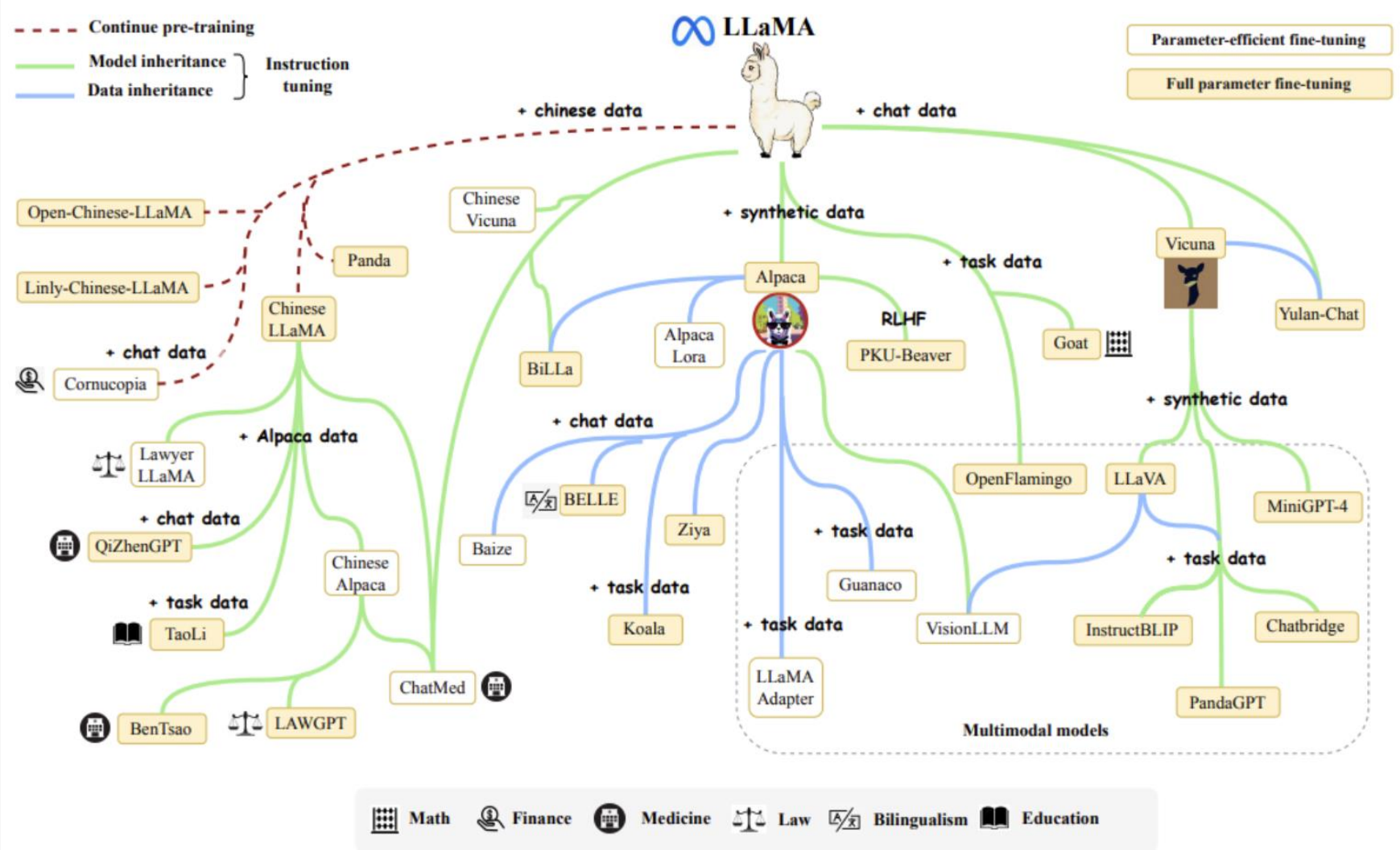
### Llama 3 & Llama 4 (Meta)

| | |
|---|---|
| Llama 3 시리즈<br>(3.1, 3.2, 3.3) | - 특징: 다양한 크기(8B ~ 405B)의 오픈 웨이트 모델<br>- 이전 아키텍처 개선, 더 크고 정제된 다국어 데이터셋 활용<br>- Llama 3.3 70B: 성능-비용 효율성 균형 |

# 거대언어모델(LLM)

## 최신 언어모델 비교 분석

### Llama 3 & Llama 4 (Meta)



## 최신 언어모델 비교 분석

### Llama 3 & Llama 4 (Meta)

| | |
|---|---|
| Llama 4 시리즈 (Scout, Maverick, Behemoth) | - 특징: Meta 최초 MoE 아키텍처, 네이티브 멀티모달 (텍스트+이미지)<br>- Scout: 17B 활성 파라미터 (109B 총), 단일 H100 실행 가능, 1000만 토큰 컨텍스트 창<br>- Maverick: 17B 활성 파라미터 (400B 총, 128 전문가), Scout 보다 고성능 목표<br>- Behemoth: 288B 활성 파라미터 (~2T 총), 교사 모델 역할<br>- 200+ 언어 훈련, iRoPE 아키텍처 (긴 컨텍스트 처리) |

## 최신 언어모델 비교 분석

### Llama 3 & Llama 4 (Meta)

| | |
|---|---|
| Llama 4 시리즈 성능 | - Llama 4 Scout: Gemma 3, Gemini 2.0 Flash-Lite 등 능가<br>- Llama 4 Maverick: 멀티모달(GPT-4o, Gemini 2.0 Flash 능가), 추론/코딩(DeepSeek-V3 경쟁)<br>- Llama 4 Behemoth: 일부 STEM 벤치마크(GPT-4.5, Claude 3.7 Sonnet, Gemini 2.0 Pro 능가) |

## 최신 언어모델 비교 분석

### Claude 3.5 & 3.7 Sonnet (Anthropic)

| | |
|---|---|
| 주요 특징 | - '유용하고(Helpful), 정직하며(Honest), 무해한(Harmless)' AI 목표<br>- Constitutional AI 기반 : AI 모델(특히 LLM)이 유용하고, 정직하며, 무해하도록 (Helpful, Honest, Harmless - HHH 원칙) 훈련하는 접근법<br>- 핵심 아이디어: 인간의 직접적인 유해성 피드백 대신, AI가 사전에 정의된 '원칙(헌법)'에 기반하여 스스로 응답을 평가하고 개선하도록 학습<br>- 안전성 및 윤리성 강조<br>- 멀티모달 (텍스트/이미지 입력 → 텍스트 출력)<br>- 20만 토큰 컨텍스트 창 |

## 최신 언어모델 비교 분석

### Claude 3.5 & 3.7 Sonnet (Anthropic)

| | |
|---|---|
| Claude 3.5 Sonnet | - 특징: 코딩, 글쓰기, 시각 데이터 추출, 에이전트 작업, 도구 사용 강점<br>- v2: '데스크 탑' 기능 추가<br>(키보드/마우스 작업 수행)<br>- 성능: 코딩, 글쓰기 등 강력한 성능 제공 |
| Claude 3.7 Sonnet | - 특징: 현재 가장 지능적인 모델, '확장된 사고(Extended Thinking)' 기능 도입 (단계별 신중 추론, 최대 64K 토큰 출력)<br>- '하이브리드 추론(hybrid reasoning)' 모델<br>- 2024년 10월까지 지식 학습 |

## 최신 언어모델 비교 분석

### Gemini 1.5 & Gemini 2.0 & Gemini 2.5 (Google)

| | |
|---|---|
| 주요 특징 | - 다양한 모델 제품군 (Nano, Flash-Lite, Flash, Pro, Ultra, Flash Thinking)<br>- 네이티브 멀티모달 (텍스트, 이미지, 오디오, 비디오, PDF 등 다양한 입력 처리)<br>- 텍스트 외 이미지/오디오 출력 지원 계획 (2.0 Flash) |

## 최신 언어모델 비교 분석

### Gemini 1.5 & Gemini 2.0 & Gemini 2.5 (Google)

| | |
|---|---|
| 주요 모델 | - Gemini 2.0 Flash: 실시간 '멀티모달 라이브 API' 제공, 향상된 에이전트, 내장 이미지/음성 생성(프리뷰)<br>- Gemini 2.0 Flash Thinking: '사고 과정' 함께 출력 (실험적)<br>- Gemini 2.5 Pro Preview: 현재 가장 강력한 추론 능력 (멀티모달 이해, 코딩, 세계 지식) |
| 특이 사항 | - 다양한 벤치마크에서 최고 수준 성능 주장, 2.5 Pro 상당한 성능 향상<br>- 초기 설계부터 멀티모달리티 핵심, 100만 토큰 컨텍스트<br>- 멀티모달 라이브 API , Thinking 모델 , Google 검색 연동 (Grounding)<br>- 관련 모델: Gemma (오픈 모델 제품군) |

## 최신 언어모델 비교 분석

### 최신 언어모델 요약

| | |
|---|---|
| LLM 발전방향 | - 멀티모달리티: 대부분 네이티브 멀티모달 기본 탑재 (텍스트, 이미지, 오디오 등)<br>- 추론 능력 강화: 전용 추론 모델 등장, 범용 모델도 추론 강조 , 추론 과정 강화 기능 (Claude 3.7 '확장된 사고')<br>- 컨텍스트 창 확장: 128K(GPT-4o) → 200K(Claude 3) → 1M(Gemini) → 10M(Llama 4 Scout)<br>- 아키텍처 혁신 (MoE): 비용 효율성 위해 Llama 4 등에서 전문가 혼합(MoE) 채택<br>- 개방성 vs. 폐쇄성: Llama/Gemma(오픈) vs GPT/Claude/Gemini API(폐쇄) 간 경쟁 구도<br>- 평가 기준 다양화: 언어 능력 외 코딩, 추론, 멀티모달, 장문맥 등으로 확장 |

# 거대언어모델(LLM)

## 최신 언어모델 활용 사례

| | |
|---|---|
| 의료 분야 | - 임상 진단 보조 (오진 감소)<br>- 의료 기록 자동 작성/전사, 가상 건강 보조원<br>- 의료 문헌 분석, 질병 조기 진단 지원 (생체 음향 분석), 영상 판독 보조 |
| 금융 분야 | - 사기 탐지, 재무 보고서 자동 생성, 시장 동향 분석<br>- AI 챗봇, 위험 평가 지원, 거래 분석 지원 (Trading GPT) |

## 최신 언어모델 활용 사례

| | |
|---|---|
| SW개발 분야 | - 코드 자동 완성/수정/리팩토링 (GitHub Copilot)<br>- 모바일 앱 자동 테스트 (Uber), AI 모델 출력 검증 (GitLab)<br>- 자연어 기반 데이터 쿼리 생성 (Honeycomb) |
| e-커머스 분야 | - 개인화 상품 추천, 검색 결과 개선 (Picnic, Swiggy)<br>- 고객 리뷰 분석 (Yelp), 이미지 기반 유사 상품 검색 (Etsy)<br>- 비정형 데이터 속성 추출 (DoorDash) |

## 최신 언어모델 활용 사례

**기타 분야**

- 고객 서비스: AI 챗봇/가상 비서 통한 24시간 응대 자동화, 문의 자동 분류 (GoDaddy),
- 마케팅/콘텐츠: 광고 문구, 블로그 등 마케팅 자료 자동 생성, 이메일 제목 생성(Nextdoor)
- 법률: 계약서 검토/분석 자동화, 법률 문서 리서치
- 교육: 맞춤형 학습 콘텐츠 생성, 자동 채점, 어학 학습 문제 생성 (Duolingo)
- 인사: 반복 HR 업무 자동화, 직원 피드백 분석
- 사이버 보안: 보안 규정 준수 점검, 악성코드 분석, 보안 사고 요약 (Google)
- 기타: 프리미엄 상품 추천 (LinkedIn), 수요 예측 (Foodpanda), 시맨틱 검색 (Expedia), 거래 질문 제안 (Digits), 채용 공고 정보 추출 (OLX), 차별적 콘텐츠 탐지 (Zillow)

**LLM 단독 사용보다 기업 데이터/워크플로우 통합 시 효과 극대화**

## LLM의 한계

**LLM의 한계**

- 환각 (Hallucinations): 사실과 다르거나 조작된 정보 생성 → 정확성 중요한 분야에서 문제
- 편향성 (Bias): 훈련 데이터의 사회적 편견 학습 및 재생산/증폭 → 불공정성, 차별 유발
- 지식 단절 (Knowledge Cutoff): 훈련 시점 이후 최신 정보 부재
- 연산 비용 (Computational Cost): 막대한 훈련/운영 자원 및 에너지 소모, 높은 비용, 환경 문제
- 일관성/신뢰성 부족: 입력 민감성, 결과 변동성, 자기 모순
- 추론 능력 한계: 복잡한 다단계 추론, 수학 문제 등 취약
- 컨텍스트 창 제한: 한 번에 처리 가능한 정보량 제한
- 프롬프트 엔지니어링 어려움: 효과적인 프롬프트 설계에 기술/노력 요구
- 보안 취약점: 악의적 프롬프트 공격(Jailbreaking, Injection)에 취약
- 진정한 이해/상식 부족: 패턴 학습 기반, 깊은 이해 부재
- 장기 기억 부재: 대화 세션 간 정보 기억 못 함

## LLM의 윤리적 문제

**LLM의 윤리적 문제**

- 편향과 불공정성: 데이터 내 편견 학습/증폭 → 차별, 불평등 심화
- 허위 정보 및 조작: 가짜 뉴스, 선전 등 대량 생성/유포 → 사회 혼란, 신뢰 훼손
- 프라이버시 침해: PII 등 민감 정보 유출 위험, 사용자 데이터 수집/활용 문제
- 저작권 및 IP: 훈련 데이터 저작권 침해 논란, 생성물 저작권 귀속 문제
- 일자리 대체 및 경제 불평등: 지식 노동 자동화 → 실업, 소득 격차 심화 우려
- 책임성 및 투명성 부족: 블랙박스 모델 → 원인 파악, 책임 규명 어려움
- 악의적 사용: 피싱, 사기, 유해 콘텐츠 확산, 악성 코드 개발 등 악용 가능성
- 환경 영향: 막대한 전력/자원 소모 → 탄소 배출, 환경 부담
- 과잉 의존 및 탈숙련화: 비판적 사고 능력 저하, 능력 과대평가 위험
- 접근성 및 디지털 격차: 기술 접근성 불평등 → 사회경제적 격차 심화

## LLM의 문제 해결방안

### 프롬프트 엔지니어링

**프롬프트 엔지니어링**

- 정의: LLM으로부터 원하는 응답을 효과적으로 이끌어내기 위해 입력 프롬프트 설계/개선 기술
- 기본 원칙: 명확성, 구체성, 맥락 제공, 역할 부여, 형식 지정, 반복 개선
- 퓨샷/원샷/제로샷: 프롬프트 내 예시 포함 여부
- 사고 연쇄 (CoT - Chain-of-Thought): 중간 추론 단계 생성 유도 → 복잡 추론 능력 향상 ("단계별로 생각해보자")
- 자기 일관성 (Self-Consistency): 여러 추론 경로 생성 → 다수결 등 통해 일관된 답변 선택
- ReAct: 추론(Thought) + 외부 상호작용(Action) + 관찰(Observation) 반복
- 사고의 트리/그래프 (ToT/GoT): 여러 추론 경로 탐색 및 평가하며 최적 해결책 모색
- 분해 (Decomposition): 복잡 문제를 하위 문제로 나누어 해결
- 자기 개선/검증 (Self-Refinement/Correction): 모델 스스로 초기 응답 비판/검증 → 개선

## LLM의 문제 해결방안

### 검색 증강 생성(RAG)

**검색 증강 생성 (RAG)**

- 개념: 응답 생성 전, 외부 신뢰성 있는 지식 베이스에서 관련 정보 검색 → 프롬프트에 추가
- 효과: 지식 단절 해결, 최신/사실 기반 응답 → 환각 감소
- 작동 방식:
  ① 외부 데이터 색인화 (벡터 DB 등), ② 사용자 쿼리 → 유사 정보 벡터 검색
  ③ 검색된 정보 + 원 쿼리 → LLM 프롬프트 전달, ④ LLM이 보강된 프롬프트로 응답 생성
- 장점: 재훈련 없이 지식 업데이트 (비용 효율적), 최신 정보 반영, 출처 제시 (신뢰도↑),
  정보 소스 제어 가능
- 고급 RAG 기법: 쿼리 확장, 재순위화, 하이브리드 검색, Self-RAG (자체 검증),
  IRCoT (검색+추론 반복) 등
- 최적화 도구: AutoRAG (최적 RAG 구성 요소 자동 탐색)

## LLM의 문제 해결방안

### 미세 조정 & PEFT

**미세 조정 & PEFT**

- 미세 조정 (Fine-Tuning): 사전 훈련 모델을 특정 작업/데이터셋에 추가 학습시켜 성능 개선
- 전체 미세 조정: 모든 파라미터 업데이트 → 많은 계산 자원 필요
- 파라미터 효율적 미세 조정 (PEFT - Parameter-Efficient Fine-Tuning):
- 개념: 전체 파라미터 중 극히 일부만 훈련 (or 추가 소규모 파라미터만 훈련)
- 효과: 전체 미세 조정과 유사 성능 + 계산/메모리/저장 비용 획기적 절감
- 맞춤형 모델 개발 기회 확대, 파국적 망각 완화
- 주요 PEFT 방법:
  * LoRA (Low-Rank Adaptation): 기존 가중치 고정, 변화량을 저랭크 행렬 곱으로 근사하여 학습 → 파라미터 수 크게 감소
  * QLoRA (Quantized LoRA): LoRA + 기존 가중치 저정밀도 양자화 → 메모리 사용량 추가 절감
  * 기타: 어댑터 튜닝, Prefix Tuning, Prompt Tuning 등

# (참고) 거대언어모델(LLM) 평가 벤치마크

## STEM 벤치마크

**개념**

- 최근 LLM(대형 언어 모델) 평가에서 자주 언급되고 있는 개념
- STEM은 과학(Science), 기술(Technology), 공학(Engineering), 수학(Mathematics)의 약자) 작성 능력

| 벤치마크 명 | 설명 |
|---|---|
| MATH (OpenAI) | 고등학교 수준 수학 문제 (~12학년), 정답 및 서술형 풀이 포함 |
| GSM8K | 초등학생 수준 수학 문제 (단계별 풀이 과정 필요) |
| ARC | AI의 과학적 상식 및 추론 능력 평가 (초등학교/중학교 수준 문제 기반) |
| MMLU (STEM subset) | 미국 대학 교양 수준의 STEM 과목 문제 테스트 |
| HumanEval | 코딩/프로그래밍 문제 해결 능력 평가 (주로 Python 코드 생성) |
| APE210K, SciQ, 등 | 과학 문제 해결 능력 평가에 특화된 데이터셋 모음 |

# (참고) 거대언어모델(LLM) 평가 벤치마크

## MMLU (Massive Multitask Language Understanding)

| | |
|---|---|
| 주요 측정 대상 | - 언어 모델의 광범위한 세계 지식 (World Knowledge)<br>- 문제 해결 능력 (Problem Solving Ability) |
| 평가 방식 | - 57개 다양한 주제(인문학, 사회과학, STEM 등) 관련 객관식 질문 활용<br>- 초급부터 전문가 수준까지 폭넓은 난이도 포함 |
| 주요 특징 | - 모델의 다방면적 지식 및 추론 능력 종합 평가<br>- LLM 일반 지능 수준 평가의 표준 벤치마크 역할 수행 |

# (참고) 거대언어모델(LLM) 평가 벤치마크

## MGSM (Multilingual Grade School Math)

| | |
|---|---|
| 주요 측정 대상 | - 언어 모델의 기초 수학적 추론 능력<br>- 다국어 환경에서의 문제 해결 능력 |
| 평가 방식 | - 초등학교 수준 수학 응용 문제를 다국어(10개 언어)로 제공<br>- 모델의 문제 이해 및 정확한 숫자 답 제출 요구 |
| 주요 특징 | - 언어와 무관한 수학적 논리 이해/적용 능력 평가<br>- 모델의 다국어 처리 능력과 추론 능력 동시 측정 가능 |

# (참고) 거대언어모델(LLM) 평가 벤치마크

## HumanEval

| | |
|---|---|
| 주요 측정 대상 | - 언어 모델의 코드 생성 능력<br>- 자연어 설명 기반 기능적으로 올바른 코드(주로 Python) 작성 능력 |
| 평가 방식 | - 164개 프로그래밍 문제(함수 시그니처, 설명, 단위 테스트 포함) 제공<br>- 모델이 함수 본문 코드 생성<br>- 생성 코드의 단위 테스트 통과 여부(pass@k)로 자동 평가 |
| 주요 특징 | - 자연어 요구사항 → 실제 작동 코드 변환 실용 능력 평가<br>- 코드 생성 능력 평가의 표준 벤치마크 역할 (Python 중심) |

# (참고) 거대언어모델(LLM) 평가 벤치마크

## STEM 벤치마크

**개념**

- 최근 LLM(대형 언어 모델) 평가에서 자주 언급되고 있는 개념
- STEM은 과학(Science), 기술(Technology), 공학(Engineering), 수학(Mathematics)의 약자) 작성 능력

| 벤치마크 명 | 설명 |
|---|---|
| MATH (OpenAI) | 고등학교 수준 수학 문제 (~12학년), 정답 및 서술형 풀이 포함 |
| GSM8K | 초등학생 수준 수학 문제 (단계별 풀이 과정 필요) |
| ARC | AI의 과학적 상식 및 추론 능력 평가 (초등학교/중학교 수준 문제 기반) |
| MMLU (STEM subset) | 미국 대학 교양 수준의 STEM 과목 문제 테스트 |
| HumanEval | 코딩/프로그래밍 문제 해결 능력 평가 (주로 Python 코드 생성) |
| APE210K, SciQ, 등 | 과학 문제 해결 능력 평가에 특화된 데이터셋 모음 |

## 거대언어모델(LLM) 탈옥(Jail Break)

| | |
|---|---|
| 개념 | - 탈옥은 주로 프롬프트 엔지니어링(Prompt Engineering) 기법을 통해 시도<br>- 사용자는 모델이 안전 제한을 인식하지 못하거나 무시하도록 교묘하게 질문이나 지시를 구성하여 공격 |
| 공격 방법 | - 역할극(Role-Playing): 모델에게 특정 역할(예: '제한 없이 무엇이든 답하는 AI')을 부여하고 그 역할에 따라 행동하도록 지시<br>- 가상 시나리오 설정: 질문을 현실이 아닌 가상의 이야기나 시나리오의 일부인 것처럼 구성<br>- 지시문 조작: 복잡하거나 모순되는 지시로 모델을 혼란스럽게 만들어 안전 필터를 우회하도록 유도<br>- 인코딩/난독화: 금지된 키워드를 직접 사용하지 않고 다른 방식으로 인코딩하거나 은닉하여 전달<br>- 접두사 주입(Prefix Injection): 특정 지침을 사용자 프롬프트 앞에 몰래 삽입.<br>- 토큰 밀반입(Token Smuggling): 눈에 잘 띄지 않는 방식으로 유해한 지시를 숨겨 전달 |

## 지시 계층(Instruction Hierarchy) 보안

| | |
|---|---|
| 개념 | - 입력된 여러 지시사항의 우선순위를 판단하는 구조<br>- 시스템 지침(system prompt)을 사용자 프롬프트보다 우선 처리<br>- 악성 프롬프트 주입에 대한 내성 강화 |
| 적용 효과 | - 프롬프트 주입 공격 방지 :  사용자가 모델의 행위 방침을 바꾸려는 시도 차단<br>- 탈옥(jailbreak) 시도 저항 : 의도된 제약을 벗어난 응답 방지<br>- 시스템 프롬프트 노출 방지 : 내부 지침 및 설정을 외부에 노출하지 않음<br>- 안정성 향상 : 민감 주제, 허위정보 대응에 효과적 |